
BENDR: USING TRANSFORMERS AND A CONTRASTIVE SELF-SUPERVISED LEARNING TASK TO LEARN FROM MASSIVE AMOUNTS OF EEG DATA.

Demetres Kostas

University of Toronto, Toronto, Canada
Vector Institute, Toronto, Canada
demetres@cs.toronto.edu

Stéphane Aroca-Ouellette

University of Toronto, Toronto, Canada
Vector Institute, Toronto, Canada

Frank Rudzicz

University of Toronto, Toronto, Canada
Vector Institute, Toronto, Canada
Li Ka Shing Knowledge Institute, Toronto, Canada

ABSTRACT

Deep neural networks (DNNs) used for brain-computer-interface (BCI) classification are commonly expected to learn general features when trained across a variety of contexts, such that these features could be fine-tuned to specific contexts. While some success is found in such an approach, we suggest that this interpretation is limited and an alternative would better leverage the newly (publicly) available massive EEG datasets. We consider how to adapt techniques and architectures used for language modelling (LM), that appear capable of ingesting awesome amounts of data, towards the development of encephalography modelling (EM) with DNNs in the same vein. We specifically adapt an approach effectively used for automatic speech recognition, which similarly (to LMs) uses a self-supervised training objective to learn compressed representations of raw data signals. After adaptation to EEG, we find that a single pre-trained model is capable of modelling completely novel raw EEG sequences recorded with differing hardware, and different subjects performing different tasks. Furthermore, both the internal representations of this model and the entire architecture can be fine-tuned to a *variety* of downstream BCI and EEG classification tasks, outperforming prior work in more *task-specific* (sleep stage classification) self-supervision.

1 Introduction

To classify raw electroencephalography (EEG) using deep neural networks (DNNs), discriminative models need to both extract useful features from raw sequences, and classify those features. This frames both the promise and the challenge of using DNNs: feature engineering could be almost entirely avoided, without introducing limitations on classifier complexity, but both feature extraction and classification need to be learned from a *limited* supply of (relevant) high-dimensional data. This challenge is evident in brain-computer interface (BCI) applications, where DNNs can struggle to determine good features. A large degree of data variability within and between different users

causes the classification performance of many model types to vary [1, 2, 3, 4]. Fundamentally, this reveals that these models lack generality, and instead rely on characteristics specific to particular subjects (and/or sessions). Furthermore, beyond these inter- and intra-personal variations, different features are relevant for different BCI tasks in the first place. Hand-selected features (sets possibly pruned later on) are distinct under different BCI paradigms, as different features better discriminate different tasks¹ [2], e.g., P300 versus motor imagery. In other words, unlike domains such as computer vision where there is a clearer understanding that nearly all DNNs tend to learn “low-level” features in earlier layers (e.g., edge-detector-like primitives) [6, 7, 8], there is no such understanding with DNNs used to process raw EEG. There are no known transferable DNN properties or operations that are easily extended to any subject, session, or task. Importantly however, the determination of which “low-level” features DNNs developed in computer vision was revealed through models that had transferable performance from general to specific tasks[6, 7]. The development of transferable DNNs for raw EEG then appears to be a promising classification tool on the one hand, but could also serve to validate existing techniques, and perhaps even suggest novel methods (if early layers do or do not correspond to existing methodologies respectively).

The difficulty of learning both “lower-level” features and an expressive classifier simultaneously may help explain why work using DNNs to classify raw BCI data has tended to prefer shallower networks [9, 10, 2, 11, 12]. With these shallower networks, the range of *learnable* features is relatively limited. By design, these employ constrained linear operations, and a limited few of these layers include subsequent non-linear activations [9], an otherwise crucial feature of DNN complexity. Fundamentally, their inability to uniformly outperform feature-engineering approaches [2] indicate that these limited features are not entirely sufficient, and more importantly, they may not always be desirable in a DNN approach [9]. In prior work we presented evidence that, if inter-personal variability had been adjusted for, the performance of shallower models more quickly saturates to lower performance levels as compared to a deeper network alternative [9], suggesting that more complex raw-BCI-trial features *could* be developed using DNNs with sufficient data, notably such that these data provide a reasonable empirical estimate of the data distribution in question. Inter/intra-person variability sabotages this approach, limiting the inter-applicability of data from all people and sessions of an entire dataset. This is disappointing since the labelling process is much more difficult than in other domains of DNN research to begin with².

In this work, we argue that self-supervised sequence learning would be an effective approach for developing and deploying more complex DNNs in BCI, as it can learn from many more people, sessions, and tasks using *unlabelled* data, thus promising to better model the input distribution of EEG data; it affords the possibility to learn features with little variability across traditionally confounding factors. Specifically, we investigate techniques inspired by language modelling (LM), that have found recent success in self-supervised end-to-end speech recognition and image recognition. We begin by comparing fully supervised transfer learning (which has been frequently looked to as an EEG/BCI TL solution) to self-supervised approaches, finding inconsistency in the extension of computer vision-style pre-training to BCI (and by extension the data domain of EEG). We then evaluate a simple adaptation of previous work in self-supervised speech recognition called wav2vec 2.0[13] to EEG. With this framework, arbitrary EEG segments are encoded as a sequence of learned feature vectors we call BErt-inspired Neural Data Representations (or ‘BENDR’). We ask whether BENDR are: transferable to novel EEG recorded from unseen subjects, different hardware, and different tasks, and if BENDR are generally suitable (both as-is or fine-tuned) to a battery of downstream EEG classification tasks.

1.1 Pre-training with DNNs

For inspiration on tackling DNN pretraining in BCI, one can look to successful applications in other domains. The modern deep learning (DL) “revolution” was ushered in on the back of computer vision and image recognition [14, 15].

¹While this is typical, some procedures, like covariance-based Riemannian classification schemes, do not necessarily need different features for different tasks [2, 5].

²Consider the difficulty of collecting and labelling 100 more BCI trials as compared to the same for 100 more images.

The successes of DL in this domain have stemmed from a lineage of massive *labelled* datasets [15], such as the ImageNet dataset [16]. These datasets were used to train deep convolutional neural networks, often one of the variants or progeny of ResNet [17] and DenseNet [18]. Crucially, these are labelled datasets, featuring – especially in the case of ImageNet – an enormous number of unique possible classification *targets* (1000 is common with ImageNet³, but more are possible⁴). As mentioned above, leveraging labelled data (especially for a particular task) of a similar scale in BCI is impractical but, despite this, a sizeable amount of prior work tries to fashion a transfer learning strategy after the successes of ImageNet pre-training. These take the form of transferring knowledge from a network trained with *more data*, typically more subjects, to a target domain with *less data*, typically a single subject [9, 19, 20, 3, 21, 22], with some work transferring between entire datasets of the same paradigm, rather than subjects [23]. On the surface, these embody a general-to-specific supervised transfer learning scheme reminiscent of ImageNet pre-training. However, these particular framings lack diversity in pre-training targets. Instead, the number and type of targets remains the same in both the pre-training and fine-tuning stages. We remain unaware of any work that pre-trains a DNN with a *wide gamut of BCI-relevant targets* to a *more narrow* target set, as would be common when using ImageNet as pre-training for more specific computer vision tasks⁵. This is noteworthy, as this is part of what makes ImageNet a *general task*. Evidence suggests that pre-training label diversity is important for effective ImageNet transfer learning [24], though an excess could be detrimental [25, 24]. More fundamentally, however, this pre-training paradigm has begun to be questioned altogether, with some work finding that it does not necessarily improve downstream performance, where commonly it has been assumed that it should (e.g., in medical images or object localization; though it *speeds up* training considerably) [26, 6, 27, 25].

What has begun to emerge as a potential alternative in computer vision – and markedly so when there is limited labelled downstream data – is self-supervised learning [28, 29, 30, 31]⁶. These works are inspired by the recent success in natural language processing (NLP) using LMs, which can be used for transfer learning, but also for few-shot and zero-shot learning [33, 34]. We propose that DNN transfer learning in BCI and neuroimaging analysis generally could follow a similar line, with *encephalography models* (EM) in place of LMs. The important question being *how best to construct such an EM, so that it learns features that are general enough while remaining usable for any analysis task?*

Prior work has developed approaches for (EEG) self-supervised sleep stage classification (SSC) through contrastive learning[35]. Contrastive learning in its most general form consists of identifying positive representations from a set that also includes incorrect or negative distractor representations [36]. Banville *et al.* proposed two potential contrastive learning tasks – a “relative positioning” task and an extension they termed “temporal shuffling”[35]. Underlying both tasks is the notion that neighbouring representations share a label. This is a fair assumption for SSC, where sleep stages change slowly, and is generally reasonable for continuous problems, where some notion of smoothness is assumed. Their proposed “relative positioning” task is a binary classification problem distinguishing whether a pair of representations are within a local or positive window τ_{pos} , or outside a long-range or negative window τ_{neg} (when $\tau_{neg} > \tau_{pos}$, those falling within τ_{neg} but outside τ_{pos} are ignored). The representations themselves are a learned mapping (in their case, a convolutional neural network) of raw EEG time-windows to a feature vector. Their alternative “temporal shuffling” method adds a third window or representation with which to contrast that is within τ_{pos} of one (arbitrary) window called the ‘anchor’, and again learns the representations through a binary classification task. In this case, the classification determines whether the three representations are ordered sequentially, or are out of order. These tasks ultimately both improved downstream SSC performance over the same network trained in a fully supervised

³image-net.org/challenges/LSVRC/2012/

⁴<http://image-net.org/about-stats>

⁵It is also worth noting that our own prior work does not consider or identify this.

⁶Terminology here can be somewhat fuzzy. What is meant by self-supervision is a supervision-like task that requires domain-relevant understanding in some sense. Sometimes, ‘semi-supervised’ is used instead, as it is often also a semi-supervised procedure [28], since the task is learned in an unsupervised fashion first and then classic supervised learning is used with labels. Typically, though, semi-supervision involves inferring labels for unlabelled data during training. Instead, self-supervision is loosely a particular case of representation learning, which is not historically uncommon in BCI [32]. Though this work is different given that typically the loss is domain or data agnostic.

manner with randomly initialized weights (with self-supervision being distinctly better when limiting fine-tuning data, a common theme in the recent wave of self-supervision literature [37, 33]) and a variant of the same network but trained under an autoencoder paradigm (alternative pretraining option; the network was pretrained to reconstruct the original waveform). “Relative positioning” performed better on average (and no statistical significance expressed) as compared to its counterpart, but a linear classification of simple hand-crafted features was still highest performing overall. These results demonstrate the promise of self-supervised learning with DNNs for EEG over a supervised approach. This is all the more valuable, as it appeared that the self-supervised time-window representations (learned features of a window), when projected into a 2D visualization, also appeared to model some sleep-stage information and information about subject age [35] purely through the contrastive task without use of sleep-stage labels. The major concern with these particular schemes though are the lengths of the time windows (τ_{pos} and τ_{neg}). The shortest windows employed were 2 minutes for τ_{pos} and τ_{neg} , which seems prohibitively long. As it is assumed that representations within τ_{pos} are similarly labelled, it may be difficult to expand the use of this technique to time scales closer to that of a BCI trial (across any paradigm), which tend to be no more than several seconds at most. Instead, we focus our efforts on adapting a relevant strategy from the wider ML literature that could develop features on smaller time scales.

Returning to transfer learning successes in NLP, the *masked* language model (MLM) is a slight variation on the typical LM which models the probability of encountering a language token given previous (or, in some cases, also subsequent) tokens. The MLM scheme instead learns to *reconstruct* language token(s) given surrounding context (fashioned after the Cloze task), and is employed by BERT [38] and its lineage [34] of similar models, which embody part of the recent wave of successful NLP transfer learning. This family of models may deploy a variety of auxiliary tasks [39] for transfer learning, but the task currently at the heart of this family is as follows: given a sequence of N tokens t_1, \dots, t_N , and a subset of token indexes I_m , for each token index $i \in I_m$, tokens are masked with some mask M so that:

$$q_i = \begin{cases} M; i \in I_m \\ t_i; \text{otherwise} \end{cases}, \forall i \in N \quad (1)$$

A transformer encoder [38, 40] then reconstructs the original sequence of tokens from the *masked* sequence (t_i and $q_i, \forall i \in N$ respectively in eq. 1). M could be a single learned token [13], or in the case of BERT: 80% of the time a fixed [MASK] token, 10% a random token or 10% the original token (with 15% of tokens masked within each sequence) [38].

Could an EM be developed in this vein, using individual samples rather than tokens (i.e., direct application of BERT to raw EEG)? Unfortunately, the highly correlated nature of neighbouring samples in EEG (or most other continuous data for that matter), is not conducive to this approach. The likely result would be that, instead of an EM, a method for interpolation would be learned, as has been argued in similar work in self-supervised learning with speech [41]. In other words, the smoothness of these data would make it hard to produce general features simply through recovering missing points. Masking a contiguous span of tokens instead, which is beneficial in NLP [42, 34], could avoid simply learning to interpolate missing samples, but the *reconstruction* of time-series data is difficult, due to the difficulty (among other things) of capturing the degree of error in time (within contiguous sequences) [43]. The losses used for such reconstruction, commonly mean squared error (or mean absolute error), erroneously assume independence in the error between elements in the series, causing inappropriate error signals when (among other things) when simply shifting a reconstruction in time [43].

Contrastive predictive coding (CPC), is a contrastive learning-based task that retains the character of sequence learning provided by masked language model-like approaches, but is not as susceptible to degeneration into interpolation, or similarly affected by the issues of time-series reconstruction [31]. With CPC, the correct *learned representation* for a particular sequence offset is predicted relative to distractor representations, typically those of other positions in the same sequence [31]. This task enables learning both a good feature representation and an understanding of the progression of

those features end-to-end. Interestingly, both the representations alone [37], and the addition of the sequence model [13] have proven potentially useful for supervised fine-tuning after pre-training.

Prior work in self-supervised speech recognition has begun to synthesize parts of CPC and MLM to produce methodologies for self-learning with raw waveforms [13, 44, 45, 41, 31]. In our work, we adapt one of these approaches called `wav2vec 2.0` [13] (its particular formulation is detailed in section 2.4.1) to EEG, and investigate how effective the representations (BENDR) are for downstream tasks.

2 Materials and methods

All experiments are implemented using the *deep neural networks for neurophysiology* (DN3) library⁷. The source code and pre-trained BENDR models can be found at <https://github.com/SPOClab-ca/BENDR>.

2.1 Datasets

The ideal pre-training dataset for our purposes would feature many subjects, each recorded over many sessions. These sessions would also ideally be distributed across large time-scales and consist of a variety of performed tasks. In other words, the pre-training dataset should consist of a representative sample of EEG data in the most general sense. This also means that these data should include multiple different recording hardware and configurations. The closest publicly accessible dataset, to our current knowledge, was the Temple University Hospital EEG Corpus (TUEG) [46]. It consists of clinical recordings using a mostly conventional recording configuration (monopolar electrodes in a 10-20 configuration) of over 10,000 people, some with recording sessions separated by as much as eight months apart. The subjects were 51% female, and ages range from under 1 years old to over 90 [46]. We focused specifically on versions 1.1 and 1.2 of this dataset which amounted to approximately 1.5 TB of European-data-format (EDF) EEG recordings *before* preprocessing.

Furthermore, we compiled a non-exhaustive battery of publicly accessible EEG data classification tasks summarized in table 1. Most of these were BCI task datasets, which could readily be compared to previous work with DNNs trained without any additional unlabelled data [9, 11]. We also included one of the sleep stage classification (SSC) tasks used by Banville *et al.* [35] in their work on sleep stage self-supervision described above, for comparison. This dataset afforded some further insight into generality, as BCI data are typically classified in the context of particular trials or events, and SSC is a more continuous problem, requiring that large spans of time are labelled with the particular sleep stage a subject is undergoing. These segments are distinctly longer than the BCI trials we considered in the remaining battery (an order of magnitude difference in our case when compared to the largest BCI task sequence length), and are distinctly closer in length to the pre-training task. This allowed us to consider how effective our approach was to such a different time-scale. Another notable difference with the SSC dataset was the scale of available labels, which seems to have enabled prior work to consider deeper and more complex models [47]. We segmented these sequences into 30 second periods as in prior work, and focused on 5 labels as in prior work [47, 35].

2.2 Preprocessing

The focus of the preprocessing stage was to create a maximally consistent representation of EEG sequences across datasets, so that the pre-trained network was well-suited to a *variety* of “downstream” tasks. More or less, this amounted to modifying downstream datasets to match the configuration of the pre-training dataset. The first aspect of this was to remove spurious differences in channel amplitude. Each sequence gathered for training was linearly scaled and shifted (a weight and offset for each sequence adjusts every sample in the sequence) so that the maximum and minimum values within each sequence equal 1 and -1 respectively. To account for the lost relative (to the entire dataset) amplitude

⁷<https://github.com/SPOClab-ca/dn3>

⁸<https://www.kaggle.com/c/inria-bci-challenge>

Dataset	Paradigm	sfreq. Hz	# Ch.	Subjects	Targets	Folds
MMI [48, 49]	MI (L/R)	160	64	105	2	5
BCIC [50]	MI (L/R/F/T)	250	22	9	4	9
ERN [51]	Error Related Negativity	200	56	26 (10)	2	4
P300 [52, 53, 49]	Donchin Speller	2048	64	9	2	9
SSC [54, 55, 49]	Sleep Staging	100	2	83	5	10

Table 1: Summary of downstream dataset battery and number of cross-validation folds used. Cross validation splits were in a leave-multiple-subjects-out configuration if $Folds < Subjects$, or leave-one-subject-out if $Folds = Subjects$ (as in prior work [9]). The ERN dataset was featured in an online competition⁸ which featured 10 held-out test subjects (not used during training), which we used as a test dataset for all four validation splits of this dataset.

information, a single channel was added with the constant value $\frac{\max(s_i) - \min(s_i)}{\max(S_{ds}) - \min(S_{ds})}$, where S_{ds} is the set of all samples in the dataset and $s_i \subset S_{ds}$ is a particular sub-sequence (i.e., trial). We additionally addressed the differences in sampling frequency and electrode sets of the different dataset. Our solutions to these problems were similarly minimalist and were achieved using standard features in DN3 [56]. Specifically, we over- or under-sampled (by whole multiples, for lower and higher sampling frequencies respectfully) to get nearest to the target sampling frequency of 256 Hz. Then, nearest-neighbour interpolation was used to obtain the precise frequency (described further in [56]). Additionally, the P300 dataset was low-pass filtered below 120 Hz to avoid aliasing due to its higher sampling rate (and associated higher original low pass filter). Furthermore, the SSC dataset featured two bi-polar electrodes: FPz-Cz and Pz-Oz, which were simply mapped to FPz and Pz, respectively. The TUEG dataset also features some higher sampling rate signals; we included those with low-pass filters that did not violate the Nyquist criterion (and subsequently re-sampled them as above), and ignored the rest.

A reduced subset of the Deep1010 channel mapping from DN3 [56] was used throughout. This ensured that particular channels were mapped to a consistent index for each loaded trial. The original mapping was designed to be more inclusive, and thus assumed up to 77 possible EEG electrodes. In the interest of minimizing unnecessary electrodes for an already high-dimensional problem, we focused on the 19 EEG channels of the *unambiguously illustrated 10/20* channel set (UI 10/20) [57], as the TUEG dataset recordings were done using a roughly 10/20 channel scheme. We simply ignored reference electrodes, electro-oculograms, and any other auxiliary channels. When also accounting for the additional relative amplitude channel described above, every sequence from every dataset used 20 channels. All surplus channels were ignored, and missing channels set to 0.

During pre-training, we extracted sequences of 60 seconds (every 60 seconds) from each usable sequence, which amounted to 15,360 samples per subsequence. We observed in early testing that there was better performance with larger sequences (see figure 2 for more). As can be seen in table 2, the downstream datasets all used sequence lengths shorter than this, but the architecture we employed (see section 2.3) was ostensibly agnostic to sequence length (see section 4 for caveats).

2.3 Model architecture

The model architecture closely follows that of wav2vec 2.0 [13] and is comprised of two stages. A first stage takes raw data and dramatically downsamples it to a new sequence of vectors using a stack of short-receptive-field 1D convolutions. The product of this stage is what we call BENDR (specifically in our case, when trained with EEG). A second stage uses a transformer *encoder* [40] (layered, multi-head self-attention) to map BENDR to some new sequence that embodies the target task.

Raw data is downsampled through the stride (number of skipped samples) of each convolution block in the first stage (rather than pooling, which would require greater memory requirements). Each of our convolution blocks comprised of

the sequence: 1D convolution, GroupNorm [58], and GELU activation [59]. Our own encoder features six sequential blocks, each with a receptive fields of 2, except for the first, which was 3. Strides matched the length of the receptive field for each block. Thus, the *effective sampling frequency* of BENDR is 96 times smaller (≈ 2.67 Hz) than the original sampling frequency (256 Hz). Each block consists of 512 filters, meaning each vector has a length of 512.

The transformer follows the standard implementation of Vaswani *et. al* [40], but with internal batch normalization layers removed and with an accompanying weight initialization scheme known as T-Fixup [60]. Our particular transformer architecture uses 8 layers, with 8 heads, model dimension of 1536 and an internal feed-forward dimension of 3076. As with wav2vec 2.0, we use GELU activations [59] in the transformer, and additionally include LayerDrop [61] and Dropout at probabilities 0.01 and 0.15, respectively, during pre-training but neither during fine-tuning. We represent position using an additive (grouped) convolution layer [13, 62] with a receptive field of 25 and 16 groups before the input to the transformer. This allows the entire architecture to be sequence-length independent, although it may come at the expense of not properly understanding position for short sequences.

Originally, the downstream target of the wav2vec 2.0 process was a downstream speech recognition *sequence* (it was fine-tuned on characters and phonemes) [13]. Instead, here the entire sequence is classified. To do this using a transformer, we adopt the common practice [38] of feeding a fixed token (*a.k.a.* [CLS] in the case of BERT or, in our case, a vector filled with an arbitrary value distinct from the input signal range, in this case: -5) as the first sequence input (prepended to BENDR). The transformer output of this initial position was not modified during pre-training, and only used for downstream tasks.

The most fundamental differences in our work as compared to that of the speech-specific architecture that inspired it are: 1. we do not quantize BENDR for creating pre-training *targets*, and 2. we have *many* incoming channels. In wav2vec 2.0, a *single* channel of raw audio was used. While a good deal of evidence [9, 63, 11, 64, 2, 12] supports the advantage of temporally-focused stages (no EEG channel mixing) separate from a stage (or more) that integrates channels, we elected to preserve the 1D convolutions of the original work to minimize any additional confound and to reduce complexity (compute and memory utilization $\propto N_{filters}$ with 2D rather than $\propto \frac{N_{filters}}{N_{EEG}}$ for 1D convolutions). This seemed fair, as there is also evidence that 1D convolutions are effective feature extractors for EEG, particularly with large amounts of data [65, 56]. Notably, wav2vec 2.0 downsampled raw audio signals by a much larger factor (320) than our own scheme, but speech information is localized at much higher frequencies than encephalographic data is expected to be. The new effective sampling rate of BENDR is ≈ 2.67 Hz, or a feature-window (no overlap) of ≈ 375 ms. We selected this downsampling factor as it remained stable (i.e., it did not degenerate to an infinite loss, or simply memorize everything immediately) during training.

2.4 Training

We used the Adam [66] optimizer throughout training, with weight decay set to 0.01. We additionally used a cosine learning rate decay with linear warmup for 5% and 10% of total training steps (batches) for pre-training and fine-tuning respectively. The peak learning rate itself varied by dataset; this, and other variable hyperparameters, are further documented in appendix A.

2.4.1 Pre-training

The pre-training procedure largely follows wav2vec 2.0, but we make some notable hyperparameter changes. Specifically, the self-supervised loss for a masked token localized at BENDR position t , is defined as:

$$\mathcal{L} = -\log \frac{\exp(\text{cossim}(c_t, b_t)) / \kappa}{\sum_{b_i \in B_D} \exp(\text{cossim}(c_t, b_i)) / \kappa} \quad (2)$$

Where c_t is the output of the transformer at position t , b_i is the BENDR vector at some offset i , and B_D is a set of 20 uniformly selected distractors from the same sequence, plus b_t . We use the cosine similarity $\text{cossim}(x, y) = x^T y / (|x||y|)$ function to determine how similar vectors are, and the sensitivity of this is adjusted by a temperature factor κ , set to 0.1. In essence, this loss operates by adjusting the output of the transformer at position t to be *most similar to the encoded representation at t , despite that this input to the transformer is masked*. We also added the mean squared activation of the BENDR to the loss, as was similarly done previously [13], but set the weight of this additional term to 1 (rather than 10).

We learn a single mask vector during pre-training of the same length as each BENDR vector, and use this as the transformer input to masked positions. Contiguous sequences of 10 are masked with probability $p_{mask} = 0.065$, such that, for each sample, the likelihood of being the *beginning* of a contiguous section was p_{mask} , and overlap is allowed. The number of negatives/distractors was set to 20 and uniformly sampled from the *same* sequence as the masked vector, i.e., negatives do not cross trials or sequences.

After pre-training, we examined how generalizable the sequence model and vectors were to unseen data, by evaluating the contrastive task, expressed as the transformer accuracy in constructing c_t to be most similar to b_t rather than the distractors. During evaluation, we masked half the amount expected during training, but such that masked spans were evenly spaced through the sequence (so that there were no overlapping sequences, and sufficient context was available). That is, for a sequence length of N_S , we masked $0.5 \times N_S \times p_{mask} = N_m$ contiguous sequences (of 10), and spaced them every $\left\lfloor \frac{N_S}{N_m} \right\rfloor$ steps (starting at the first sample). N_S first remained at 15, 360 (60 seconds as in training, no overlap between subsequent sequence representations) for all datasets except P300, where sessions were too short and instead 5120 (20 seconds) was used. We then evaluated the change in performance across the downstream datasets, excluding P300, as N_S varied from 20-60 seconds.

2.4.2 Downstream fine-tuning

Ultimately, our aims for subject-, session-, and dataset-generalizable representations were not simply to accurately mimic the correct input, but with the intent that these representations – and potentially the sequence model itself – could be effectively transferred to specific and arbitrary tasks. We considered six different variations of TL across the battery of EEG classification tasks presented in table 1:

1. Add a new softmax classification layer to the first (pre-pended position) output token of the transformer and train the entire model to classify the downstream targets.
2. Ignore the pre-trained transformer, average pool the BENDR to four concatenated vectors, add a new classification layer and train the model (only the first stage and new layer) to classify the downstream targets.
3. The same as (1.), but without pre-training
4. The same as (1.), but keep the BENDR fixed and continue training the transformer.
5. The same as (2.), but without pre-training
6. The same as (2.), but keep the first stage weights fixed and train only the new classification layer.

We considered these permutations so that we could speak to the effect each stage had on downstream performance, at least to some degree. First, we were interested in 1) determining whether the new sequence representation (BENDR) contained valuable features *as-is* (as they appear to for speech [13]) or if they required some further training, and 2) whether the sequence model learned characteristics of the BENDR that were informative to the classification task. Finally, ignoring pre-training all-together, of course, was to examine how effective the network would be at learning the task otherwise, without pre-training or transfer learning.

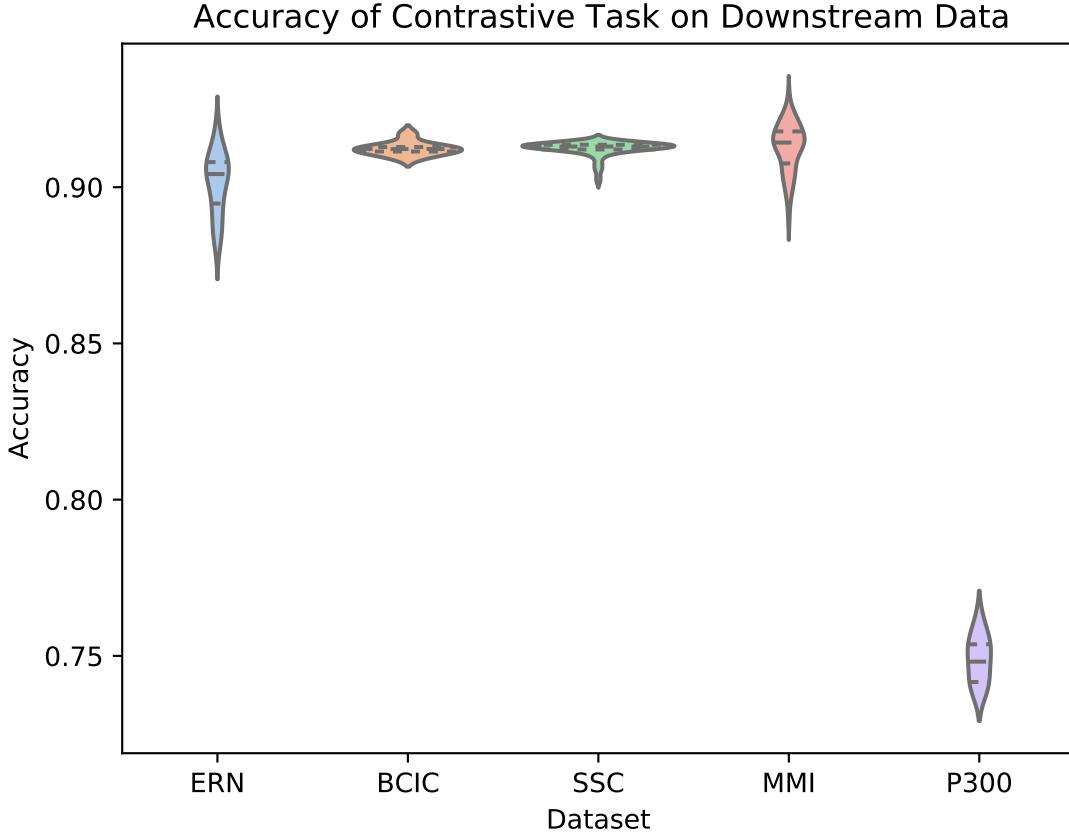


Figure 1: Violin plot (inner lines for quartile divisions) of test-subject-wise accuracy for each downstream dataset. Specifically, accuracy of the sequence model (transformer stage) at creating a representation that is closest to the correct representation at masked sequence positions. The P300 dataset is distinctly lower performing than the remaining datasets, though this was likely due to its shorter evaluation context (see figure 2). Nonetheless, there is minimal test-subject-wise variation, particularly when compared to classifier performance generally.

The P300, ERN, and SSC datasets all had imbalanced class distributions; we adjusted for these imbalances by *undersampling* points of the more frequent classes with replacement so that the number of samples drawn – per epoch – of each class was equal to the number of examples of the least frequent target class.

We also included the sequence regularization proposed by wav2vec 2.0 [13], though we adjusted it for our more varied trial lengths. That is, in all 6 fine-tuning configurations, contiguous sections of 10% of the entire BENDR of a trial were masked with the mask token learned during pre-training (not changed after pre-training) at a probability of 0.01. In other words, this was the likelihood of a sample being the beginning of a contiguous masked section, as in pre-training. Additionally across the BENDR (throughout each vector in the sequence), a similar procedure dropped features to 0, where contiguous sections of 10% of the channels (51) were dropped with a probability of 0.005.

3 Results

3.1 Pre-training generalization

Figure 1 shows how accurate the transformer stage is at producing an appropriately similar BENDR when compared to distractor representations. There are two key observations in this figure, the first is that there is little variability across the

Dataset	Start (s)	Length (s)	Metric	Best	Model config.
MMI	0	6	BAC	86.7	Linear (2.)
BCIC	-2	6	Accuracy	42.6	Linear (2.)
ERN	-0.7	2	AUROC	0.65	Linear (2.)
SSC	0	30	BAC	0.72	Linear (2.)
P300	-0.7	2	AUROC	0.72	BENDR (1.)

Table 2: Performances of downstream datasets. Start and length refer to length of trials and start with respect to event markers in seconds. Best performance specifies average performance across all subjects (and therefore folds) for best performing model configuration. BAC: class balanced accuracy; AUROC: area under the receiver operating characteristic curve. Model configurations are numbered in accordance with the list presented in section 2.4.2.

first four datasets, and within each of the five datasets. The latter point implies that this accuracy is not radically variable across different subjects, as it tends to be when considering classifier performance [1, 3] (though, when fine-tuning for classification, this variability returns; see figure 3). This could be because a) the transformer adequately learns a general model of how BENDR sequences of novel persons and equipment progressed, b) the BENDR themselves are invariant to different people, hardware, and tasks, c) some combination of the last two possibilities, or d) the problem is being solved via some non-signal characteristics. We return to this question shortly. The second observation was alluded to already: the P300 dataset distinctly under-performs the other downstream datasets. However, this coincided with the shortest evaluation sequence. Looking at figure 2, we see that all five datasets have consistently similar performance when evaluated with 20 seconds of data, so the dip in P300 performance of figure 1 seems less remarkable. Taken together, 1 and 2 clearly indicate that a longer evaluation context makes the contrastive task easier. This suggests that the contrastive task is, in fact, solved by learning signal-relevant features, rather than some more crude solution like interpolation, or by simply creating a sequence of recognizable position representations (both of which have no reason to exhibit this dependence on sequence length). We believe the most likely explanation for the rise in performance with more context is that local representations are more difficult distractors, implying that the new effective sampling rate remains too high (and there is still redundant information encoded in local BENDR). Notwithstanding, there is a strong uniformity of performance across datasets and subjects (in both figures 1 and 2), meaning this scheme develops features (whether through the transformer itself, or the BENDR) that generalize to novel subjects, hardware, and tasks, though their applicability to downstream contexts remains to be seen.

3.2 Downstream fine-tuning

Figure 3 and table 2 present a picture of how effectively BENDR could be adapted to specific tasks. Overall, the fine-tuned linear classification (listed as downstream configuration 2. above) that bypassed the transformer entirely after pre-training was highest performing four out of five times, though using the transformer for classification (1.) performed consistently similarly (confidence intervals always overlapped), and surpassed the bypassed transformer (2.) with the P300 dataset (and was highest performing for this dataset). Deploying the full network (initial stage and transformer) without pre-training was generally ineffective, though this was not the case with the SSC dataset, which may have been due to the larger data availability. In fact, for both the full and linear model architectures trained with the SSC data, fine-tuning the pre-trained model is mostly on par with the randomly initialized counterpart. Considering our results with the SSC data relative to those of Banville *et. al.*'s [35] proposed contrastive learning for sleep staging (described in section 1.1), their reported results show that the fine-tuned variants of our own model (1. and 2.) achieved a higher mean balanced accuracy relative to their two proposed schemes. Taken in concert with our own approach's wider applicability and more fine-grained temporal feature development, we believe this demonstrates that ours is a promising alternative. Interestingly, with and without pre-training (2. and 5.) achieved similar performance to Banville *et. al.*'s fully supervised results (where our configurations and their architecture employ similar 1D convolution-based schemes),

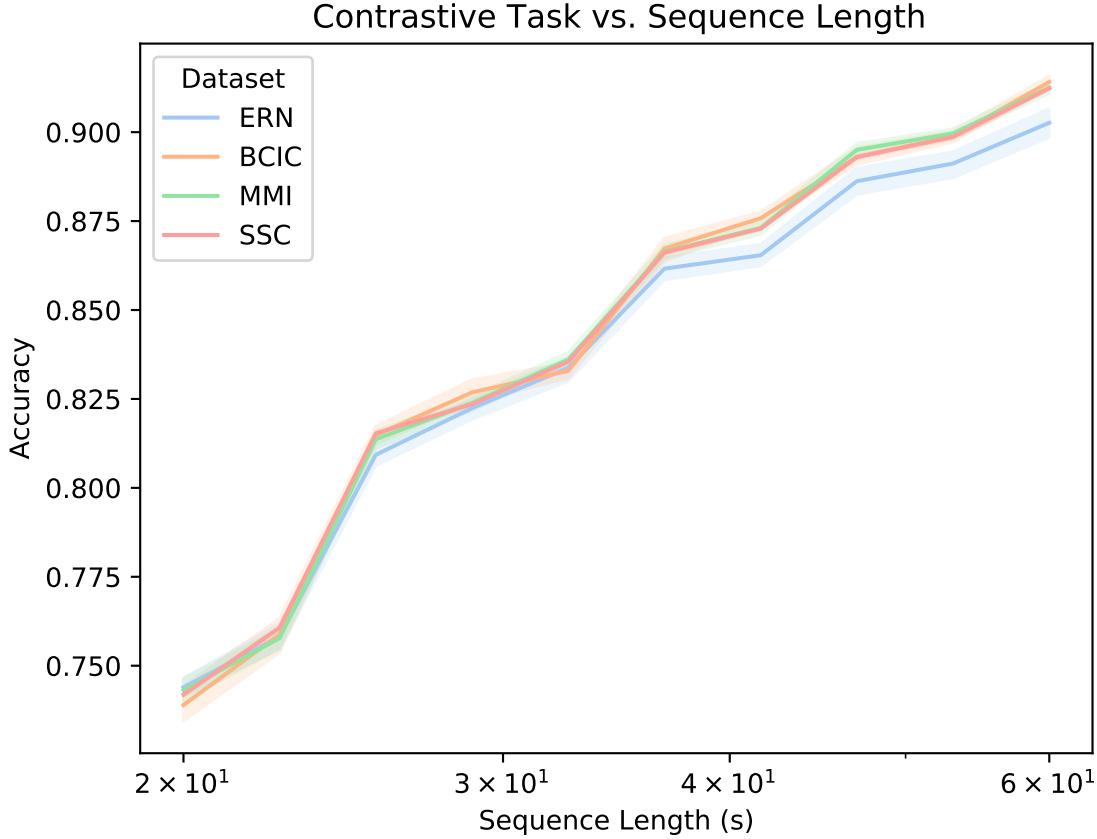


Figure 2: Contrastive accuracy versus evaluation length in seconds (x-axis logarithmic). Performance is distinctly similar for all datasets, rising for longer sequences. We suggest that this implies that samples that are further apart are easier to distinguish between than neighbouring samples. Thus, while BENDR encode local signal characteristics well, there is redundancy.

which is notable as with this dataset, both their “temporal-shuffling” and “relative-positioning” tasks under-performed full supervision when utilizing the full SSC dataset.

Our fine-tuned approaches similarly appear reasonably competitive with prior work on the MMI dataset [9, 3], particularly when considering that only 19 channels (rather than the full set of 64) were being used. In all considered configurations, despite heavy regularization (and the very low learning rates) the randomly initialized parameters were consistently prone to overfitting, all the more so with the full model architecture. Conversely, the pre-trained networks were slow to fit to the downstream training data (under the exact same training scheme for fine-tuning). Ultimately, though most of these results are not necessarily state-of-the-art, this single pre-training scheme nonetheless shows a breadth of transferability which is apparently unique.

4 Discussion

We are unaware of any prior work assessing transformer-based [40] DNNs with EEG data (raw or otherwise). This is perhaps consistent with the ineffectiveness we observed with the randomly initialized full architecture (3.) and could imply that effective use of this powerful emerging architecture *requires* pre-training (or at least enough data, given the

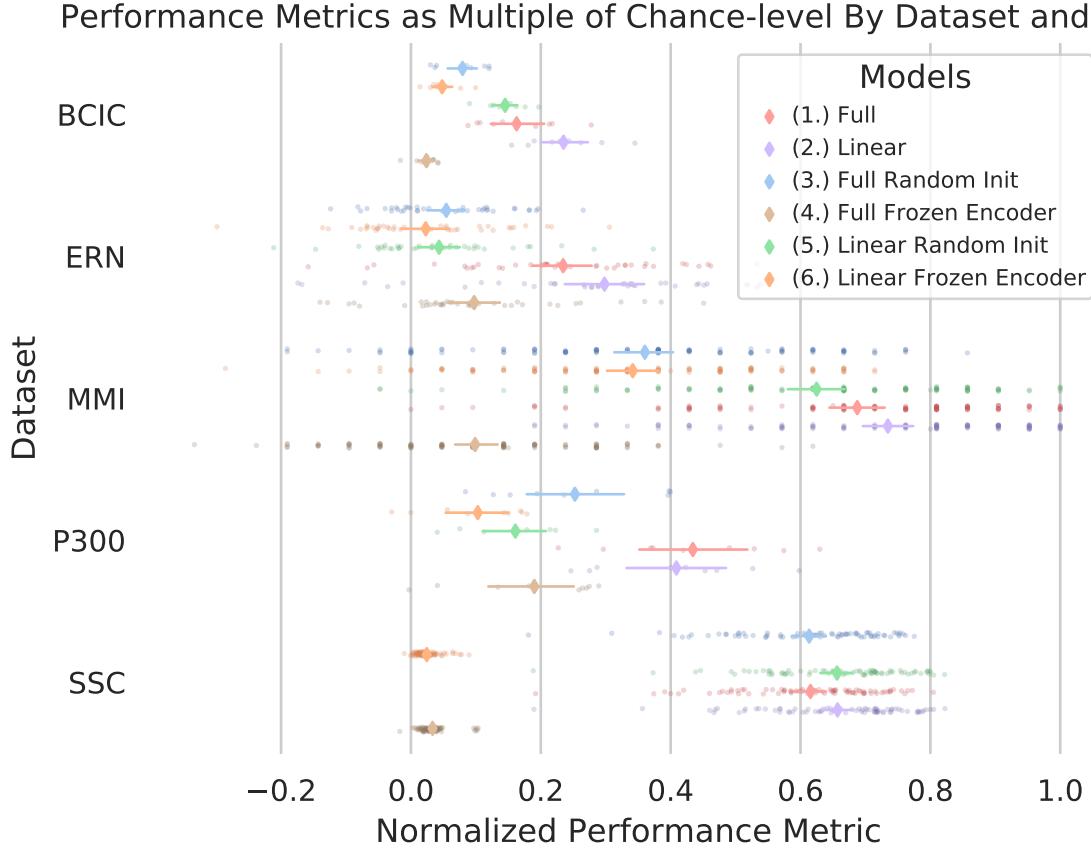


Figure 3: Performance of all downstream datasets for each of the six model configurations considered. Metrics vary by dataset, see table 2. Metrics were normalized to range from chance (0) to perfect (1). Individual translucent points are performances of single subjects (within each test fold), solid diamonds indicate mean performance across all subjects/folds, with surrounding bars showing .95 confidence intervals using $n = 1000$ bootstrap sampling. The discretized pattern of the MMI dataset is due to the limited trials *per subject*, which resulted in limited distribution of performance levels. Notably here, (1.) or (2.) were consistently among the best performing, yet both remained within the confidence levels of each other. The randomly initialized average-pooled BENDR with linear classifier (5.) also performed well, though less consistently. Model configurations are numbered in accordance with the list presented in section 2.4.2.

better looking SSC performance). Future work should continue to evaluate this architecture, particularly as it appears to be more widely applicable than the NLP applications it was originally proposed for [67, 13].

We believe that our approach can be improved through adjusting the neural network architecture and pre-training configuration such that it becomes more data-domain (EEG) appropriate. Future work will prioritize effective integration of spatial information, likely by better isolating temporal and spatial operations. Evaluation using large downstream datasets that *also* feature many channels, such as the *Montreal Archive of Sleep Studies* (MASS)⁹ will be considered. Though available for public access at the time of writing, these data were unavailable while experiments were prepared and conducted. Prior work shows that DNN approaches effective for EEG leverage spatial information [64], and it is presently unclear to what degree this is the case with BENDR. In terms of data-appropriate temporal modelling, which we have considered with relatively more zeal in this work, recall that figure 2 presents the possibility that local representations may be retaining redundant information, further improvements may be found in better compressing the

⁹<http://massdb.herokuapp.com/en/>

temporal resolution of BENDR. Future work will consider larger downsampling factors in the initial stage, along with longer sequences, balancing the more difficult problem of summarizing more data (in effect, further data *compression*), with the apparent increased effectiveness of the contrastive task (as observed in figure 2) on longer sequences. A small but potentially fruitful avenue for further improvement includes reconsidering the additive convolutional layer as a substitute for explicit position encodings, which are in fact more common [34, 38, 40]. Recall that this was originally for two reasons: `wav2vec 2.0` did the same, and we felt it best to limit excessive changes to the architecture on a first iteration, and because it seamlessly supported flexible input lengths. This latter point comes, however, with a trade-off – our particular position encoder had a receptive field of 25 (stride of 1), which means a little over 9 seconds of input. While it seems that convolutional position encodings offer better performance [62], this input width exceeded the *entire* length of all but the sleep classification task (the length we chose was optimized for pre-training behaviour).

After considering these possible avenues for improving BENDR, we still do not fully discount the validity of some of the transfer learning paths we appear to exclude above in our introduction. We will reconsider these paths in future work. Particularly, given the success we had in crossing boundaries of hardware in this work, and in prior work [56], it may be possible to construct an *aggregate* dataset featuring a variety of EEG classification tasks, towards better ImageNet-like pre-training. The construction of a more coherent label set that crosses several BCI paradigms would no doubt be a significant effort (e.g., problems may include: is a rest period before one task paradigm the same as rest before another? What about wakeful periods in sleep?). This would no doubt be imbalanced; the labels would be distributed in a long-tailed or Zipfian distribution that would likely require well thought-out adjustment [68, 69]. Furthermore, the value of ImageNet pre-training *seems to be* localized to very early layers and the internalization of domain-relevant data statistics [70, 6]. Future work could look into which of these may be leveraged with a new aggregate (multiple subjects *and* tasks) pre-training, or the common subject-specific fine-tuning. This may provide insight into better weight initialization, or integration of explicit early layers similar to [6] (one could also argue that SincNet layers [71] are some such layers that could factor here). Additionally, as temporally-minded reconstruction losses continue to develop [43], reconsidering the effectiveness of signal reconstruction as a pre-training objective (and/or regularization) is warranted, whether this is within an MLM-like scheme similar to BENDR, or a seq2seq model [72].

5 Conclusion

We have proposed MLM-like training as a self-supervised pre-training step for BCI/EEG DNNs. This is in the interest of diversifying the investigations into successful transfer learning schemes for DNNs applied to BCI and EEG. While previous approaches fashioned DNN transfer learning after ImageNet pre-training, we find this approach inadequate as there is limited applicable data availability and it is questionably analogous to its forebear. While our proposed alternative might similarly suffer from this latter point to some degree (the most distinct MLM success is with discrete sequences, not continuous ones), it is more conducive to leveraging potentially immense amounts of unlabelled data, it is not limited to long-term feature developments as with previous proposals, and it seems to produce representations equally suited to different users and sessions, which is a problem ImageNet pre-training appears less suited to solving. In summary, we see strong paths for the effective deployment of powerful computation and massive data scales with EEG and BCI. Effective solutions in these specific applications could help drive application *and* analysis solutions in neuroimaging and perhaps physiology generally.

References

- [1] Claudia Sannelli, Carmen Vidaurre, Klaus-Robert Müller, and Benjamin Blankertz. A large scale screening study with a SMR-based BCI: Categorization of BCI users and differences in their SMR activity. *PLOS ONE*, 14(1):e0207351, 1 2019.

- [2] F Lotte, L Bougrain, A Cichocki, M Clerc, M Congedo, A Rakotomamonjy, and F Yger. A review of classification algorithms for EEG-based brain-computer interfaces: a 10 year update. *Journal of neural engineering*, 15(3):031005, 2018.
- [3] Hauke Dose, Jakob S. Møller, Helle K. Iversen, and Sadasivan Puthusserypady. An end-to-end deep learning approach to MI-EEG signal classification for BCIs. *Expert Systems with Applications*, 114:532–542, 2018.
- [4] Minkyu Ahn and Sung Chan Jun. Performance variation in motor imagery brain-computer interface: A brief review. *Journal of Neuroscience Methods*, 243:103–110, 2015.
- [5] Paolo Zanini, Marco Congedo, Christian Jutten, Salem Said, and Yannick Berthoumieu. Transfer Learning: A Riemannian Geometry Framework with Applications to Brain-Computer Interfaces. *IEEE Transactions on Biomedical Engineering*, 65(5):1107–1116, 2018.
- [6] Maithra Raghu, Chiyuan Zhang, Jon Kleinberg, and Samy Bengio. Transfusion: Understanding transfer learning for medical imaging. *arXiv*, (NeurIPS), 2019.
- [7] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding Neural Networks Through Deep Visualization. 6 2015.
- [8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS’12, pages 1097–1105, USA, 2012. Curran Associates Inc.
- [9] Demetres Kostas and Frank Rudzicz. Thinker invariance: Enabling deep neural networks for BCI across more people. *Journal of Neural Engineering*, 17(5):56008, 2020.
- [10] Yannick Roy, Hubert Banville, Isabela Albuquerque, Alexandre Gramfort, Tiago H Falk, and Jocelyn Faubert. Deep learning-based electroencephalography analysis: a systematic review. *Journal of Neural Engineering*, 16(5):051001, 2019.
- [11] Vernon J. Lawhern, Amelia J. Solon, Nicholas R. Waytowich, Stephen M. Gordon, Chou P. Hung, and Brent J. Lance. EEGNet: A compact convolutional neural network for EEG-based brain-computer interfaces. *Journal of Neural Engineering*, 15(5):aace8c, 2018.
- [12] Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for EEG decoding and visualization. *Human Brain Mapping*, 38(11):5391–5420, 11 2017.
- [13] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. 2020.
- [14] Terrence J. Sejnowski. The unreasonable effectiveness of deep learning in artificial intelligence. *Proceedings of the National Academy of Sciences*, page 201907373, 1 2020.
- [15] Yann LeCun, Yoshua Bengio, Geoffrey Hinton, Lecun Y., Bengio Y., and Hinton G. Deep learning. *Nature*, 521(7553):436–444, 2015.
- [16] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.
- [17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *Multimedia Tools and Applications*, pages 1–17, 12 2015.
- [18] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q Weinberger. Densely Connected Convolutional Networks. *CoRR*, 8 2016.

- [19] Fatemeh Fahimi, Zhuo Zhang, Wooi Boon Goh, Tih-Shi Lee, Kai Keng Ang, and Cuntai Guan. Inter-subject transfer learning with an end-to-end deep convolutional neural network for EEG-based BCI. *Journal of Neural Engineering*, 16(2):026007, 2019.
- [20] Gaowei Xu, Xiaoang Shen, Sirui Chen, Yongshuo Zong, Canyang Zhang, Hongyang Yue, Min Liu, Fei Chen, and Wenliang Che. A Deep Transfer Convolutional Neural Network Framework for EEG Signal Classification. *IEEE Access*, 7:112767–112776, 2019.
- [21] Michael A. Schwemmer, Nicholas D. Skomrock, Per B. Sederberg, Jordyn E. Ting, Gaurav Sharma, Marcia A. Bockbrader, and David A. Friedenberg. Meeting brain–computer interface user performance expectations using a deep neural network decoding framework, 2018.
- [22] Yuan-Pin Lin and Tzzy-Ping Jung. Improving EEG-Based Emotion Classification Using Conditional Transfer Learning. *Frontiers in Human Neuroscience*, 11(June):1–11, 2017.
- [23] Apiwat Ditthapron, Nannapas Banluesombatkul, Sombat Ketrat, Ekapol Chuangsawanich, and Theerawit Wilaipr-asitporn. Universal Joint Feature Extraction for P300 EEG Classification Using Multi-Task Autoencoder. *IEEE Access*, 7:68415–68428, 2019.
- [24] Minyoung Huh, Pulkit Agrawal, and Alexei A. Efros. What makes ImageNet good for transfer learning? *CoRR*, pages 1–10, 2016.
- [25] Jiquan Ngiam, Daiyi Peng, Vijay Vasudevan, Simon Kornblith, Quoc V. Le, and Ruoming Pang. Domain adaptive transfer learning with specialist models. *arXiv*, 2018.
- [26] Kaiming He, Ross Girshick, and Piotr Dollar. Rethinking imageNet pre-training. *Proceedings of the IEEE International Conference on Computer Vision*, 2019-Octob(i):4917–4926, 2019.
- [27] Simon Kornblith, Jonathon Shlens, and Quoc V. Le. Do better imagenet models transfer better? In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2019-June, pages 2656–2666, 2019.
- [28] Kan Chen, Jiang Wang, Liang-Chieh Chen, Haoyuan Gao, Wei Xu, and Ram Nevatia. ABC-CNN: An Attention Based Convolutional Neural Network for Visual Question Answering. *arXiv*, 2016.
- [29] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning. 200, 2020.
- [30] Olivier J. Hénaff, Ali Razavi, Carl Doersch, S. M. Ali Eslami, and Aaron Van Den Oord. Data-efficient image recognition with contrastive predictive coding, 2019.
- [31] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation Learning with Contrastive Predictive Coding. 2018.
- [32] Xiang Zhang, Lina Yao, Xianzhi Wang, Jessica J. M. Monaghan, David Mcalpine, and Yu Zhang. A survey on deep learning-based non-invasive brain signals: recent advances and new frontiers. *Journal of Neural Engineering*, 2020.
- [33] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. *CoRR*, 5 2020.
- [34] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li Peter, and J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2019.

- [35] Hubert Banville, Isabela Albuquerque, Aapo Hyvarinen, Graeme Moffat, Denis-Alexander Engemann, and Alexandre Gramfort. Self-Supervised Representation Learning from Electroencephalography Signals. In *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 10 2019.
- [36] Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning. In *36th International Conference on Machine Learning, ICML 2019*, volume 2019-June, pages 9904–9923, 2019.
- [37] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big Self-Supervised Models are Strong Semi-Supervised Learners. *arXiv*, (NeurIPS):1–18, 2020.
- [38] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, 2018.
- [39] Stéphane Aroca-Ouellette and Frank Rudzicz. On Losses for Modern Language Models. pages 4970–4981, 2020.
- [40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. 2017.
- [41] Dongwei Jiang, Wubo Li, Ruixiong Zhang, Miao Cao, Ne Luo, Yang Han, Wei Zou, and Xiangang Li. A Further Study of Unsupervised Pre-training for Transformer Based Speech Recognition. 5 2020.
- [42] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. SpanBERT: Improving Pre-training by Representing and Predicting Spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 7 2020.
- [43] Francois Rivest and Richard Kohar. A New Timing Error Cost Function for Binary Time Series Prediction. *IEEE Transactions on Neural Networks and Learning Systems*, 31(1):174–185, 2020.
- [44] Alexei Baevski and Abdelrahman Mohamed. Effectiveness of Self-Supervised Pre-Training for ASR. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7694–7698. IEEE, 5 2020.
- [45] Yu-An Chung, Hao Tang, and James Glass. Vector-Quantized Autoregressive Predictive Coding. In *Interspeech 2020*, volume arXiv, pages 3760–3764, ISCA, 10 2020. ISCA.
- [46] Iyad Obeid and Joseph Picone. The temple university hospital EEG data corpus. *Frontiers in Neuroscience*, 10(May), 2016.
- [47] Sajad Mousavi, Fatemeh Afghah, and U. Rajendra Acharya. SleepEEGNet: Automated sleep stage scoring with sequence to sequence deep learning approach. *PloS one*, 14(5):e0216456, 2019.
- [48] Gerwin Schalk, Dennis J Mcfarland, Thilo Hinterberger, Niels Birbaumer, Jonathan R Wolpaw, and A Brain-computer Interface B C I Technology. BCI2000 : A General-Purpose Brain-Computer Interface (BCI) System. *IEEE Transactions on Biomedical Engineering*, 51(6):1034–1043, 2004.
- [49] Ary L Goldberger, L. A. Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, C. K. Peng, and H Eugene Stanley. PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23), 2000.
- [50] Michael Tangermann, Klaus Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J. Miller, Gernot R. Müller-Putz, Guido Nolte, Gert Pfurtscheller, Hubert Preissl, Gerwin Schalk, Alois Schlögl, Carmen Vidaurre, Stephan Waldert, and Benjamin Blankertz. Review of the BCI competition IV. *Frontiers in Neuroscience*, 6(JULY):1–31, 2012.
- [51] Perrin Margaux, Maby Emmanuel, Daligault Sébastien, Bertrand Olivier, and Mattout Jérémie. Objective and Subjective Evaluation of Online Error Correction during P300-Based Spelling. *Advances in Human-Computer Interaction*, 2012:1–13, 2012.

- [52] Luca Citi, Riccardo Poli, and Caterina Cinel. Documenting, modelling and exploiting p300 amplitude changes due to variable target delays in donchin's speller. *Journal of Neural Engineering*, 7(5):056006, sep 2010.
- [53] Luca Citi, Riccardo Poli, and Caterina Cinel. Erp-based brain-computer interface recordings, 2014.
- [54] Bastiaan Kemp, Aeilko H. Zwinderman, Bert Tuk, Hilbert A.C. Kamphuisen, and Josefien J.L. Oberyé. Analysis of a sleep-dependent neuronal feedback loop: The slow-wave microcontinuity of the EEG. *IEEE Transactions on Biomedical Engineering*, 47(9):1185–1194, 2000.
- [55] Bastiaan Kemp, Aeilko Zwinderman, Bert Tuk, Hilbert Kamphuisen, and Josefien Oberyé. The sleep-edf database [expanded], 2018.
- [56] Demetres Kostas and Frank Rudzicz. Dn3: An open-source python library for large-scale raw neurophysiology data assimilation for more flexible and standardized deep learning. *bioRxiv*, 2020.
- [57] Valer Jurcak, Daisuke Tsuzuki, and Ippeita Dan. 10/20, 10/10, and 10/5 systems revisited: Their validity as relative head-surface-based positioning systems. *NeuroImage*, 34(4):1600–1611, 2007.
- [58] Yuxin Wu and Kaiming He. Group Normalization. *International Journal of Computer Vision*, 128(3):742–755, 2020.
- [59] Dan Hendrycks and Kevin Gimpel. Gaussian Error Linear Units (GELUs). pages 1–9, 2016.
- [60] Xiao Shi Huang, Felipe Perez, Jimmy Ba, and Maksims Volkovs. Improving transformer optimization through better initialization. *Proceedings of Machine Learning and Systems 2020*, pages 9868–9876, 2020.
- [61] Angela Fan, Edouard Grave, and Armand Joulin. Reducing transformer depth on demand with structured dropout. *arXiv*, 103:1–15, 2019.
- [62] Abdelrahman Mohamed, Dmytro Okhonko, and Luke Zettlemoyer. Transformers with convolutional context for ASR. *arXiv*, 4 2019.
- [63] Demetres Kostas, Elizabeth W Pang, and Frank Rudzicz. Machine learning for MEG during speech tasks. *Scientific Reports*, 9(1):1609, 12 2019.
- [64] Stanislas Chambon, Mathieu N. Galtier, Pierrick J. Arnal, Gilles Wainrib, and Alexandre Gramfort. A Deep Learning Architecture for Temporal Sleep Stage Classification Using Multivariate and Multimodal Time Series. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(4):758–769, 2018.
- [65] Lukas A.W. Gemein, Robin T. Schirrmeister, Patryk Chrabaszcz, Daniel Wilson, Joschka Boedecker, Andreas Schulze-Bonhage, Frank Hutter, and Tonio Ball. Machine-learning-based diagnostics of EEG pathology. *NeuroImage*, 220, 10 2020.
- [66] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pages 1–15, 2015.
- [67] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. pages 1–21, 2020.
- [68] Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. Long-Tailed Classification by Keeping the Good and Removing the Bad Momentum Causal Effect. *(NeurIPS)*:1–12, 2020.
- [69] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in Neural Information Processing Systems*, 32(NeurIPS):1–18, 2019.
- [70] Behnam Neyshabur, Hanie Sedghi, and Chiyuan Zhang. What is being transferred in transfer learning? *arXiv*, (NeurIPS):1–12, 2020.
- [71] Mirco Ravanelli and Yoshua Bengio. Interpretable Convolutional Filters with SincNet. *Arxiv*, (Nips), 11 2018.

- [72] Alex Graves. *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer, c2012, Berlin ; New York, 2012.

A Downstream hyperparameters

Dataset	Batch Size	Epochs	Learning Rate
MMI	4	7	1×10^{-5}
BCIC	60	15	5×10^{-5}
ERN	32	15	1×10^{-5}
P300	80	20	1×10^{-5}
SSC	64	40	5×10^{-5}

Table 3: Hyperparameters that varied between datasets, these were not changed between different model configurations (see list in section 2.4.2).

CONCEPT-BASED EXPLAINABILITY FOR AN EEG TRANSFORMER MODEL

Anders Gjølbye Madsen^{†}*

Áshildur Jónsdóttir^{}*

William Theodor Lehn-Schiøler^{†}*

Bergdís Arnardóttir^{}*

Lars Kai Hansen^{}*

^{*}Technical University of Denmark

Department of Applied Mathematics and Computer Science
2800 Kgs. Lyngby, Denmark

[†]BrainCapture

2800 Kgs. Lyngby, Denmark

ABSTRACT

Deep learning models are complex due to their size, structure, and inherent randomness in training procedures. Additional complexity arises from the selection of datasets and inductive biases. Addressing these challenges for explainability, Kim et al. (2018) introduced Concept Activation Vectors (CAVs), which aim to understand deep models' internal states in terms of human-aligned concepts. These concepts correspond to directions in latent space, identified using linear discriminants. Although this method was first applied to image classification, it was later adapted to other domains, including natural language processing. In this work, we attempt to apply the method to electroencephalogram (EEG) data for explainability in Kostas et al.'s BENDR (2021), a large-scale transformer model. A crucial part of this endeavour involves defining the explanatory concepts and selecting relevant datasets to ground concepts in the latent space. Our focus is on two mechanisms for EEG concept formation: the use of externally labelled EEG datasets, and the application of anatomically defined concepts. The former approach is a straightforward generalization of methods used in image classification, while the latter is novel and specific to EEG. We present evidence that both approaches to concept formation yield valuable insights into the representations learned by deep EEG models.

Index Terms— Explainable AI, EEG Concepts, TCAV, BENDR

1. INTRODUCTION

We investigate representations of electroencephalogram (EEG) data obtained by self-supervised learning methods. Self-supervision is motivated by the lack of labeling in large-scale EEG datasets as labeling is both time-consuming and requires highly specialised EEG expertise. Self-supervised models, such as BERT-inspired Neural Data Representations (BENDR) [1], have the potential to overcome this challenge by learning informative representations from raw, unlabeled data. Such models can subsequently be fine-tuned for downstream classification tasks. We apply the Testing Concept Activation Vectors (TCAV) approach of Kim et al. [2], an interpretability method introduced in 2018, to BENDR-based models, to provide insights into their structure and decision-making processes. See Figure 1 for a conceptual overview. A better understanding of EEG transformer models using TCAV could support the use of these models as diagnostic support tools for identifying EEG abnormalities, such as seizures. However, the question that arises is, what constitutes human-friendly concepts in this context? To address this, we present the following scientific contributions:

This work is supported by The Pioneer Centre for AI, DNRF grant number P1, The Novo Nordisk Foundation grant NNF22OC0076907 "Cognitive spaces - Next generation explainability", and travel grants from The Danish Data Science Academy awarded to AGM and WLS.

- The first TCAV workflows for EEG data, proposing concepts based on human-annotated data as well as concepts defined by human anatomy and EEG frequency ranges.
- Sanity checks for TCAV to ensure valid explanations in simple EEG settings.
- Two practical applications: seizure prediction and brain-computer interfacing.

All code used in this research, along with references to the datasets, have been made publicly accessible for validation and replication¹.

2. THEORY

2.1. BERT-inspired Neural Data Representations

BENDR [1] is inspired by language modeling techniques that have found success also outside text analysis, in self-supervised end-to-end speech recognition and image recognition. It aims to develop EEG models for better brain-computer interface (BCI) classification, diagnosis support, and other EEG-based analyses. Importantly, the approach being based on self-supervision can learn from any EEG data using only unlabeled data. The main goal of BENDR is to create self-supervised representations with minimal robust to context boundaries like datasets and human subjects. The approach is expected to be transferable to future unseen EEG datasets recorded from unseen subjects, different hardware, and different tasks. It can be used as-is or fine-tuned for various downstream EEG classification tasks.

The architecture is based on wav2vec 2.0 [3] developed for speech processing and consists of two stages. The first stage takes raw data, and down-samples it using a stack of short-receptive field 1D convolutions, resulting in a sequence of vectors called BENDR. The second stage uses a transformer encoder [4] to map BENDR to a new sequence related to the target task. Down-sampling is achieved through strides, and the transformer follows the standard implementation with some modifications. The entire sequence is then classified, with a fixed token implemented as the first input for downstream tasks [5]. BENDR differs from the speech-specific architecture in two ways: (1) BENDR is not quantized for pre-training targets, and (2) it has many incoming channels, unlike wav2vec 2.0 which uses quantization and is based on a single channel of raw audio. The 1D convolutions are preserved in BENDR, to reduce complexity. We note that BENDR down-samples at a lower factor than wav2vec 2.0, here resulting in an effective sampling rate of ≈ 2.67 Hz equivalent to a feature window of ≈ 375 ms.

2.2. Linear Head BENDR

For downstream fine-tuning, we use a version where the pre-trained transformer modules are ignored, such that the pre-trained convolu-

¹<https://github.com/AndersGMadsen/TCAV-BENDR>

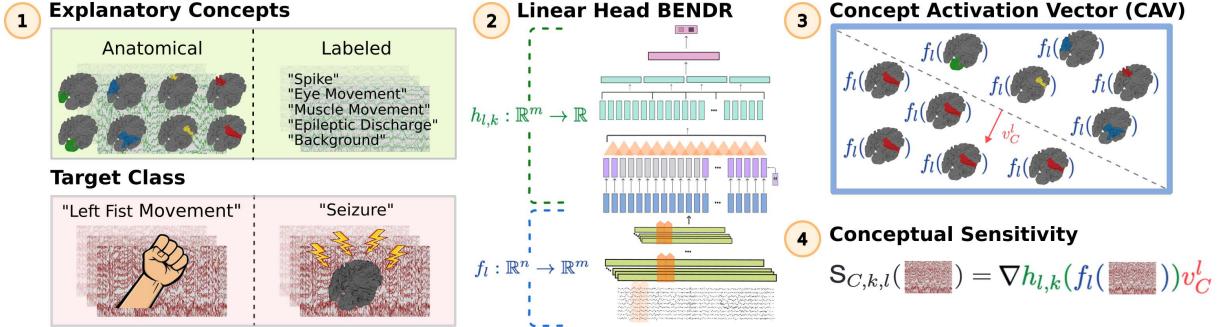


Fig. 1. An overview of using the TCAV method for EEG classification tasks with the Linear Head BENDR model: (1) Explanatory concepts are defined as either event-based EEG labels or frequency-based cortical activity, (2) Layer activations are extracted from a fine-tuned Linear Head BENDR, (3) Concept Activation Vectors (CAV) are defined as the normal vector to the hyperplane separating layer activations for concept data from those of random examples, and (4) The sensitivity of class data for a specific bottleneck of a concept is defined as the directional derivative in the direction of the respective CAV.

tional BENDR stage is used as representation, see [1]. A consistent-length representation is created by dividing the BENDRs into four contiguous sub-sequences, averaging each sub-sequence, and concatenating them. A new linear layer with softmax activation is added to classify the downstream targets based on this concatenated vector of averaged BENDR. We call this the Linear Head BENDR (LHB) model and the structure is illustrated in Figure 2.

The final LHB architecture consists of the following components:

1. **Feature encoder:** Fine-tunes the pre-trained parameters and uses six convolution blocks, each containing a temporal convolution, group normalization, and a GELU activation function to produce a BENDR of length 512.
2. **Encoding augment:** Involves masking and contextualizing the BENDR, with 10% of the BENDR masked and 10% of the channels dropped, while relative positional embeddings from the pre-trained task are added to the BENDR and further preprocessed.
3. **Summarizer:** Applies adaptive average pooling to create four contiguous sub-sequences, averaging each sub-sequence to ensure the model's independence from the input length of EEG recordings.
4. **Extended classifier:** Flattens the four sub-sequences, passes them through a fully connected layer to reduce their dimension, applies a dropout layer, uses a ReLU activation function, and normalizes the output using batch normalization.
5. **Classifier:** Consists of a linear layer with a softmax activation function, which performs the classification task.

2.3. Testing with Concept Activation Vectors (TCAV)

Testing with Concept Activation Vectors (TCAV) is a technique used to quantify the degree to which layers of neural networks align with human-defined concepts [2]. The method is general in the sense that it is not confined to the particular structure of the network nor to the data type. In its essence, TCAV can be broken down into five steps

First, the process involves defining human-aligned concepts and representing them in the data. Alongside these, data from the target class must also be present for evaluation purposes. Furthermore, to establish the directions of the concept activation vector in the latent space, it is necessary to have a collection of concept-negative or random examples.

Second, the layer activations of the concept input and the random input, respectively, are collected and separated by training a binary

linear classifier. Then, the concept activation vector, v_c^l is defined as the normal vector to the hyperplane that separates the two classes (concept vs. random).

Third, for a layer l in the network, the directional derivatives for the target class k along the learned activation vector for concept C is used to calculate how sensitive the prediction of the network is to changes in the input data in the direction of C . We can quantify the sensitivity by

$$S_{C,k,l}(\mathbf{x}) = \nabla h_{l,k}(f_l(\mathbf{x})) \cdot v_C^l, \quad (1)$$

where $h_{l,k}$ is defined as the function that maps activations in layer l through the remaining network and predicts class k .

Fourth, computing the sensitivity for several target examples, $\mathbf{x} \in X_k$, the TCAV score is defined as the ratio of examples that have positive sensitivity, i.e.,

$$\text{TCAV}_{C,k,l} = \frac{|\{\mathbf{x} \in X_k : S_{C,k,l}(\mathbf{x}) > 0\}|}{|X_k|}. \quad (2)$$

In this way, concept activation vectors that are positively aligned with target activations have a TCAV score close to 1 and concept activation vectors that are negatively aligned with target activations have a TCAV score close to 0.

Fifth and final, collecting samples of TCAV scores over several training runs, a suitable statistical test is used to assess the statistical significance of concept activation vectors aligning with the activation of target examples. The null hypothesis of the test is that half of the examples have positive sensitivity and the other half have negative or zero sensitivity, i.e.,

$$H_0 : \text{TCAV}_{C,k,l} = 0.5. \quad (3)$$

Concepts C for which the null hypothesis is rejected thus relate to the target class prediction, and may bring positive or negative evidence for the given target k .

2.4. Source localization

Source localization for EEG data involves mapping electrical signals recorded on the scalp surface to corresponding regions on the cortical surface of the brain. This process uses a head model and the EEG data collected from electrodes placed on the scalp. The reconstruction is a grid of dipolar sources. The solution to this ill-posed problem is called the lead field and there exist many different

Linear Head BENDR

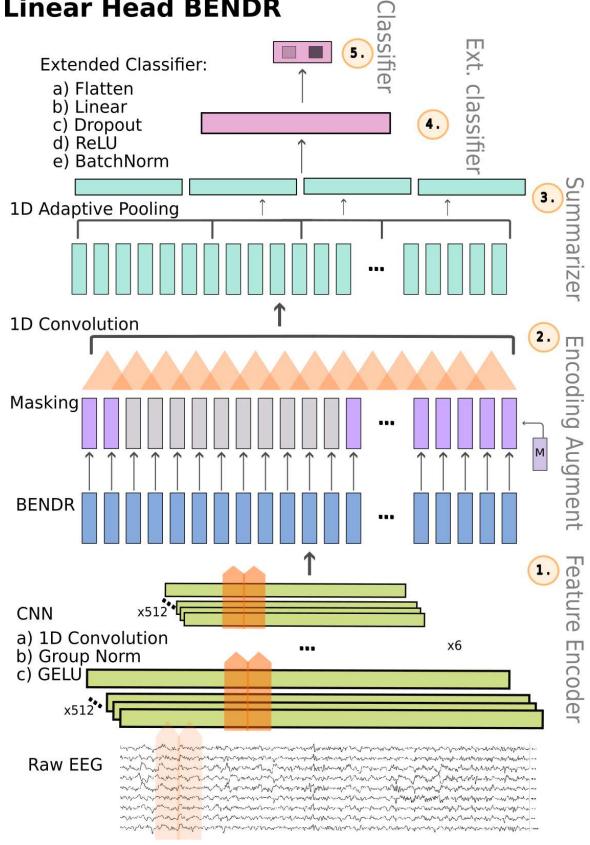


Fig. 2. The Linear Head BENDR (LHB) model architecture illustrated. The model consists of (1) Feature encoder of six confrontational blocks, (2) Encoding augment comprised of masking and convolutional contextualizer, (3) Summarizer using Adaptive Average Pooling, (4) Extended Classifier for dimensionality reduction, and (5) Classifier.

ways to obtain this solution. In this work, we use the exact low-resolution electromagnetic tomography (eLORETA) method implemented in the MNE library [6].

The eLORETA approach presupposes that the EEG measurements of the electric field present on the scalp reflect dipolar sources located in the cerebral cortex. These are conceptually modeled as a three-dimensional distribution of dipoles. The spatial resolution of eLORETA is relatively coarse, which can make pinpointing exact cortical sources challenging. However, for our purpose of estimating aggregated source activity over broadly defined brain regions, such reduced resolution is not an issue.

3. METHODS

3.1. Data

EEG is a non-invasive technique to record the brain's electrical activity. EEG data in this paper refers to these measurements, used often in research and healthcare to identify neurological conditions. In this work, we use five publicly accessible datasets, namely TUH EEG Corpus [7], TUH EEG Artifact (TUAR) Corpus, TUH EEG Events (TUEV) Corpus, TUH EEG Seizure (TUSZ) Corpus [8] and the EEG Motor Movement/Imagery (MMIDB) Dataset [9].

The TUH EEG Corpus contains 69,652 clinical and unlabeled EEG recordings obtained from Temple University Hospital (TUH).

The TUH EEG Artifact Corpus, a labeled subset of the TUH EEG Corpus, includes annotations for five distinct artifacts including eye movement artifact (*eyem*). The TUEV is a subset of the TUH EEG Corpus and includes annotations of event-based EEG segments. There are numerous categories, but we primarily focus on five key classes: (1) technical artifacts (*artf*), (2) background (*bckg*), (3) generalized periodic epileptiform discharge (*gped*), (4) periodic lateralized epileptiform discharge (*pled*), and (5) spike and slow wave (*spsw*). The TUSZ contains EEG signals with manually annotated data for seizure events.

The MMIDB EEG dataset consists of data from 109 participants who are performing or imagining specific motor tasks; our main interest is the moments when subjects either close or imagine closing their left or right fist following a visual cue. We are excluding participants S088, S090, S092, and S100 due to missing data, resulting in 105 participants.

In the construction of brain anatomy concepts, it is imperative to obtain an extensive collection of resting-state EEG data. Due to the limited availability of public datasets with the requisite size and reliability, we utilized The TUH EEG Corpus and source localization to develop a dedicated anatomically labeled resting-state dataset. A set of predefined criteria were employed, including the number of EEG channels, minimum duration, minimum sampling frequency, scaling, and the exclusion of extreme values, which led to the elimination of approximately 90% of the initial EEG recordings. Following this, a manual examination of a part of the remaining data was performed, ultimately yielding 200 human-verified resting-state EEG recordings, corresponding to an aggregate of about 70 hours of EEG data.

In the process of downstream fine-tuning and concept formation, we employ 19 EEG channels, namely *Fp1*, *Fp2*, *F7*, *F3*, *Fz*, *F4*, *F8*, *T7*, *C3*, *Cz*, *C4*, *T8*, *T5*, *P3*, *Pz*, *P4*, *T6*, *O1*, and *O2* (see the MNE documentation [6] for more information). These channels originate from the initial pre-training of BENDR using The TUH EEG Corpus. In instances where the datasets lack these channels, we establish the following mapping: *T3* \mapsto *T7*, *T4* \mapsto *T8*, *P7* \mapsto *T5*, and *P8* \mapsto *T6*. We also resample the corresponding EEG data to a 256 Hz sampling frequency and apply a high-pass FIRWIN filter with a 0.1 Hz cutoff, a low-pass FIRWIN filter with a 100.0 Hz cutoff, and a 60 Hz FIRWIN notch filter to eliminate powerline noise. In situations where preprocessing cannot be performed, the EEG is excluded. Finally, we scale each trial to the range $[-1, 1]$ and append a relative amplitude channel, see [1], resulting in a total of 20 channels.

3.2. Training

Pre-training of BENDR is based on the large set of unlabelled EEG data from The TUH EEG Corpus. The pre-training procedure is largely based on wav2vec 2.0 and involves two main stages: The convolutional stage and the transformer stage. The convolutional stage generates a sequence of representations (BENDRs) that summarize the original input. This sequence is then fed into the transformer stage, which adjusts its output to be most similar to the encoded representation at each position. The layers affected during pre-training are the feature encoder and the transformer. Kostas et al. [1] kindly made the pre-trained weights of the encoder and contextualizer publicly available, and this is the model that we have employed here.

The LHB model architecture described in Figure 2 is used for downstream fine-tuning. We aim to optimize the model for two distinct binary classification objectives. First, the model is fine-tuned for the differentiation between *seizure* and *non-seizure* events, using

the TUSZ Corpus with 60-second window segments. The hyperparameters are determined using Bayesian optimization to maximize the validation F_1 -score. The fine-tuning employs a batch size of 80, a learning rate of 1×10^{-4} , and 30 epochs. This results in a model with a balanced accuracy of 0.73 ± 0.07 .

In our second fine-tuning example, the model is adapted for the differentiation between *Left Fist Movement* versus *Right Fist Movement*, using the MMIDB EEG Dataset with 4-second window segments. We are using both the imaginary and performed task data from the 105 participants. We train the model for 7 epochs with a batch size of 4 and a learning rate of 1×10^{-5} . The hyperparameters were chosen based on the best validation balanced accuracy from leave-one-subject-out cross-validation where the model was trained for 50 epochs and the best model was retained. The specific hyperparameter configuration aligns with the optimal hyperparameters found by the original authors [1] and we find a similar balanced accuracy of 0.83 ± 0.02 .

3.3. Constructing Concepts

To construct human-aligned explanatory EEG concepts, a number of initial investigations were conducted. The data processing involved follows the methodology previously mentioned. In this section, we provide a general pipeline overview and discuss several choices made throughout the process.

Concepts from Labeled EEG Data: Using the labeled EEG data from the TUAR and TUEV Corpus and the MMIDB EEG Dataset, we create concepts representing activities within specific time windows. Each annotated segment of the EEG data is divided into windows of predetermined length and assigned the corresponding label.

In the TUEV Corpus, we define concepts for the spike/short wave (*spsw*), periodic lateralized epileptic discharge (*pled*), general period epileptic discharge (*gped*), technical artifact (*artf*), and background (*bckg*) with 60-second windows. This approach aligns with the length of the *seizure* classifier.

Lastly, we examine the eye movement (*eyem*) from the TUAR Corpus and *Left Fist Movement* and *Right Fist Movement* from the MMIDB EEG Dataset, both using 4-second windows. These different-sized windows then constitute examples of concepts defined based on their labels.

Anatomical Concepts from Unlabeled EEG Data: The objective is to identify concepts representing specific frequency bands within distinct areas of the cortex, e.g. *alpha activity in pre-motor cortex* or *gamma activity in early visual cortex*. To obtain a non-task-specific representation of each cortical area, we utilize resting-state EEG data, as it spontaneously generates activity throughout the cortex. For this purpose, we use a subset of The TUH EEG Corpus, as described above.

To define anatomical concepts, EEG data is segmented into 4-second windows, with the first and last 5 seconds of each sequence excluded to minimize artifact contamination. The data is then divided into five frequency bands with a FIRWIN bandpass filter: *delta* (0.5-4Hz), *theta* (4-8Hz), *alpha* (8-12Hz), *beta* (12-30Hz), and *gamma* (30-70Hz). The inverse operator for the forward model is computed using eLORETA [6] via the MNE Python library. Since the spatial resolution is not critical, minimal regularization of 1×10^{-4} is applied.

Using the combined version of the multi-modal parcellation of the human cerebral cortex, HCPMMP1 [10] and the inverse operator, the average power of electrical activity in 23 cortical areas for each hemisphere is determined.

Our interest lies in cortical areas exhibiting the greatest deviation

from typical activity within a specific frequency band. However, cortical areas are not equidistant from the scalp or consistent in baseline activity across bands. To normalize for these differences in the distribution of cortical activity, we compute the mean and standard deviation of the power in each cortical area for each frequency band on an EEG session level, which will be employed in various ways. We call these the baseline mean and the baseline standard deviation.

We explore possible approaches to how the baseline means and standard deviation for each EEG session could be used to normalize the power of 4-second windows within that session. The options include dividing by the baseline standard deviation to account for scalp source variation, subtracting or dividing by the baseline mean to identify the cortical area with the greatest deviation, taking the absolute difference or not, and selecting a single cortical area across all frequency bands or only within a specific band.

Identifying a single frequency and cortical area for each 4-second window of EEG data is a challenging task without prior work to guide the process, and each method presents its own limitations. We specifically look for *alpha* desynchronization in the cerebral cortex during imagined or actual movement and closed or open eyes in the MMIDB EEG dataset, i.e., that *alpha* activity in cortical areas decreases when activated. Using a paired t-test to examine the presence of lateralization in cortical activities for different methods, we found that the preferred approach is to choose the area which maximizes the absolute difference between the given time window's power and the baseline mean, divided by the baseline standard deviation, only within specific frequency bands.

Random Concepts: Construction of CAVs calls for data examples that are considered random with respect to the concept of interest. In all experiments, random concepts consisting of 4-second or 60-second windows were randomly sampled from resting-state data obtained from the subset of the TUH EEG Corpus and unannotated sections of the TUAR dataset.

3.4. Experiments

We investigate two approaches for defining explanatory concepts in EEG data. The TCAV method is then employed to evaluate whether the LHB model uses specifically defined human-aligned concepts of EEG data. For all concepts, the resulting activation vectors for all five bottlenecks in the LHB model architecture are examined to determine if they significantly align with the latent representations of class data in the model. We conduct the following experiments:

1. **Sanity Checks:** We verify the TCAV method and construction of concepts function as intended through a series of sanity checks when classifying *Left Fist Movement*.
2. **Event-based Concepts:** We assess whether the LHB model leverages specific EEG events in the classification of *seizure*.
3. **Anatomy/Frequency-based Concepts:** We investigate if the LHB model employs lateralization in cortical activity in the *alpha* band for classifying *Left Fist Movement*. The chosen cortical areas are based on their relevance to the classification task.

In the experiments, we use the TCAV method with a regularized linear model and stochastic gradient descent (SGD) learning, setting the regularization parameter $\alpha = 0.1$ to learn the decision boundary between explanatory and random concepts. We employ 50 random concepts and a maximum of 40 examples per concept. These parameters were chosen to increase statistical power. The mean TCAV scores for the target concept examples and the random examples are compared using the non-parametric Mann-Whitney U Rank test, as

opposed to the t-test used in the original TCAV method, as we observed a clear violation of the normality assumption for the TCAV scores. To mitigate Type I errors, the p-values are corrected for each experiment employing the conservative Bonferroni method, after which we claim significance if the corrected p-value is below 0.05.

4. RESULTS

4.1. Sanity Checks

We first provide evidence that the TCAV method can be applied to explain EEG data and the LHB model. In Figure 3, the high significance of class data as concepts (*Left Fist Movement* with positive evidence and *Right Fist Movement* with negative evidence) confirms this. Furthermore, concepts based on maximal activity in either the left or right hemisphere for the *alpha* frequency band strongly indicate that lateralized cortical activity is detected by several layers in the model, as expected.

Moreover, the negative alignment of a concept based on labeled artifacts with the model representation of motor task data implies that artifacts in EEG data significantly influence classification tasks. We find that *eyem* has a negative impact on the classification of *Left Fist Movement*. Note that this does *not* mean that *eyem* positively affects the opposite class, that is *Right Fist Movement*, as the TCAV Score is specific to the "*Left Fist Movement* dataset". Conversely, *eyem* could negatively affect the classification of both *Left Fist Movement* and *Right Fist Movement*, due to the lower signal-to-noise ratio for classification when artifacts are present.

4.2. Event-based concepts

We next investigate whether fine-tuning the LHB model for seizure classification on the TUSZ dataset and using explanatory concepts defined with labeled data from TUEV aligns with the model's internal representation for data labeled as containing seizures. The target of the investigation is the *seizure* label and we test all bottlenecks in the LHB model. The results of this experiment are shown in Figure 4.

When compared to EEG data labeled as containing seizures, the epilepsy-related concepts *pled*, which is present in certain brain areas, and *gped*, which is present in most of the brain, exhibit high and positive evidence in nearly all bottlenecks. This observation

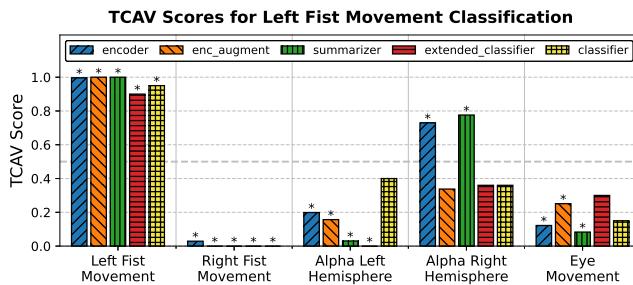


Fig. 3. Sanity checks for applying the TCAV method to EEG data and the bottlenecks of the LHB model. The figure presents the results of TCAV for the *Left Fist Movement* class in a binary classification task using the MMIDB EEG dataset. From right to left, concepts are defined as follows: (1) *Left Fist Movement* and (2) *Right Fist Movement* class data, maximal mean activity in the alpha frequency band for (3) *Left Hemisphere* and (4) *Right Hemisphere*, respectively, and (5) *Eye Movement* artifacts. Stars indicate either positive (a score above 0.5) or negative (a score below 0.5) statistical significance.

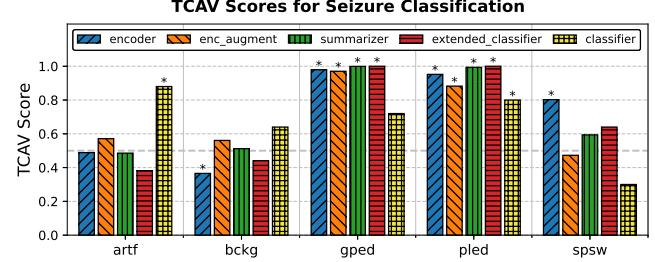


Fig. 4. The results of utilizing TCAV to assess whether event-based EEG labels align with the internal representation of the *seizure* class data in the LHB model at the five bottlenecks are presented. From the right, the concepts are defined as (1) technical artifacts (*artf*), (2) background (*bckg*), (3) generalized periodic epileptic discharge (*gped*), (4) periodic lateralized epileptic discharge (*pled*), and (5) spike and short wave (*spsw*). Stars indicate either positive (a score above 0.5) or negative (a score below 0.5) statistical significance.

aligns with existing literature that associates epileptiform discharges with seizures [11], and it is expected that the LHB model will use these properties for classification. The *spsw* concept also demonstrates significant positive evidence in the *encoder* bottleneck but not in the further downstream bottlenecks. Similarly, the *bckg* concept shows negative evidence in the *encoder* bottleneck but not in the further downstream bottlenecks. It is interesting that these concepts only come to be significant in the initial bottleneck. A possible explanation is that the technical artifacts *artf* and *bckg* are not significant for the classification, but BENDR effectively identifies seizure-related concepts and filters out noise. The results also suggest that the model's *classifier* and *extended classifier* can be further optimized, as *artf* is near-significant level in these bottlenecks and, as a result, the noise has not been completely removed. In conclusion, these examples indicate that concept-based explainability can provide valuable model design information.

4.3. Anatomy/Frequency-Based Concepts

We have demonstrated that labeled EEG data can generate human-aligned concepts, which are integrated into the LHB model for seizure classification. This comes quite naturally as labeled data is labeled by humans and tend to align with human-relatable concepts. We then present evidence that defining explanatory concepts based on cortical activity in frequency bands may uncover patterns corresponding to the model's internal representations.

In particular, for a motor classification task using the MMIDB EEG dataset and targeting the *Left Fist Movement* class, we show that cortical activity in the *alpha* band aligns with the model's internal representation. In Figure 5, we find that the CAV for *Somatosensory and Motor Cortex* in the right hemisphere positively aligns with the activations of *Left Fist Movement* class data across all bottlenecks in the model. The mean TCAV scores are also consistently positively significant. At the same time, the TCAV scores for the same cortical area in the *Left Hemisphere* are either negatively significant or insignificant. These results strongly suggest that the model's internal representation incorporates lateralization, reflecting the fact that one hemisphere exhibits more electrical activity than the other. It is noteworthy that lateralization is most significant in the *Encoding Augment* and *Summarizer* bottlenecks, indicating that it is captured early in the network.

Additionally, we observe that the *Primary Visual Cortex (V1)* areas do not exhibit lateralization, and their TCAV scores are insignificant across all bottlenecks and for both hemispheres. This further

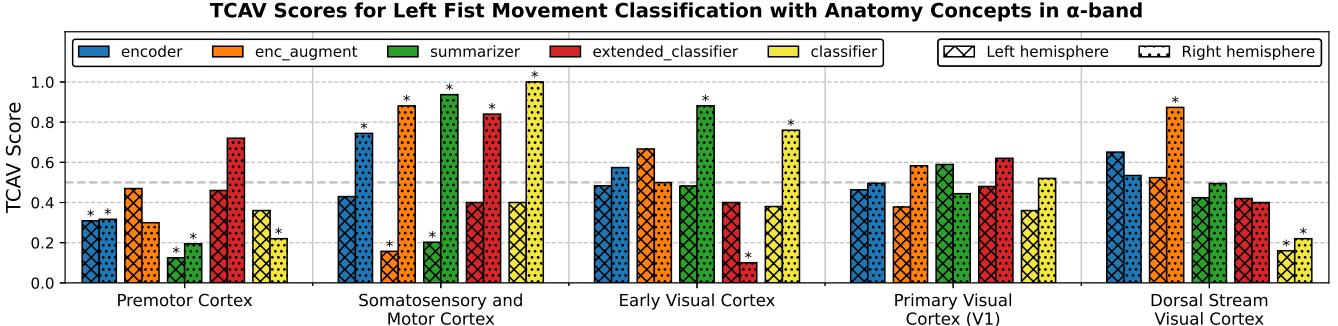


Fig. 5. Using TCAV, we analyzed the alignment between anatomical concepts in the *alpha* band and the internal representation of the *Left Fist Movement* class in the LHB model at five bottlenecks. The visualization of five pairs of concepts focused on five cortical areas, located in both the left and right hemispheres, that were deemed most relevant for the classification task. The chosen concepts had a higher deviation in the *alpha* band. Stars indicate either positive (a score above 0.5) or negative (a score below 0.5) statistical significance. Our analysis reveals significant lateralization in the *Somatosensory and Motor Cortex* across all five bottlenecks. Additionally, we observe that the *Primary Visual Cortex (VI)* was insignificant for both hemispheres in all bottlenecks.

supports the conclusion that the LHB model utilizes specific cortical areas in its classification rather than all areas indiscriminately.

While no apparent lateralization is present in the *Premotor Cortex*, this part of the cortex is negatively significant in the *Encoder* and *Summarizer* bottlenecks for both the left and right hemispheres. A possible explanation is that the instances we examine involve participants *performing* movements; therefore, there may not necessarily be relevant activity in the *Premotor Cortex*, which is primarily involved in movement planning [12].

Lastly, we observe significance in the *Classifier* bottleneck for *Early Visual Cortex* and *Dorsal Stream Visual Cortex*. We note that the movement is activated by a visual cue; however, further experiments would be required to fully clarify the effect.

5. CONCLUSION

Concept-based explainability has proven to be valuable in various domains, such as image classification and natural language understanding, where concepts are naturally defined using labeled data. In this study, we have explored the definition of concepts for EEG models for the first time. We presented two new workflows for concept-based explainability within the TCAV framework for EEG data. First, we adopted an approach akin to the original work of Kim et al. [2], in which concepts are derived from labeled data. In this case, we utilized various annotated EEG databases, e.g., data from the Temple University Hospital EEG database. The second workflow is based on the source location of resting-state EEG data also from the Temple University Hospital database. This enables us to generate datasets for TCAV derived from anatomical brain areas and for specific frequency bands, e.g., the *alpha* band. We demonstrated a proof of concept through several "sanity check" experiments to verify expected responses in elementary EEG settings, such as EEG lateralization during left- or right-hand movement. Lastly, we examined two practical applications: A case study involving seizure prediction, where TCAV reveals the role of fundamental spike patterns, and a brain-computer interface case, hinting at how the TCAV method can assist in debugging and offer valuable insights into classifier design for EEG data.

6. REFERENCES

- [1] Demetres Kostas, Stéphane Aroca-Ouellette, et al., "Bendr: Using transformers and a contrastive self-supervised learning

task to learn from massive amounts of eeg data," *Frontiers in Human Neuroscience*, vol. 15, 2021.

- [2] Been Kim, Martin Wattenberg, et al., "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav)," 2018.
- [3] Alexei Baevski, Henry Zhou, et al., "wav2vec 2.0: A framework for self-supervised learning of speech representations," *CoRR*, vol. abs/2006.11477, 2020.
- [4] Ashish Vaswani, Noam Shazeer, et al., "Attention is all you need," 2017.
- [5] Jacob Devlin, Ming-Wei Chang, et al., "BERT: Pre-training of deep bidirectional transformers for language understanding," Minneapolis, Minnesota, June 2019, pp. 4171–4186, Association for Computational Linguistics.
- [6] Roberto D. Pascual-Marqui, Dietrich Lehmann, et al., "Assessing interactions in the brain with exact low-resolution electromagnetic tomography," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 369, no. 1952, pp. 3768–3784, 2011.
- [7] Amir Harati, Silvia Lopez, et al., "The tuh eeg corpus: A big data resource for automated eeg interpretation," 12 2014.
- [8] Vinit Shah, Eva von Weltin, et al., "The temple university hospital seizure detection corpus," *Frontiers in Neuroinformatics*, vol. 12, 2018.
- [9] Gerwin Schalk, Dennis J. McFarland, et al., "Bci2000: a general-purpose brain-computer interface (bci) system," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 1034–1043, 2004.
- [10] Matthew F. Glasser, Timothy S. Coalson, et al., "A multi-modal parcellation of human cerebral cortex," *Nature*, vol. 536, no. 7615, pp. 171–178, Aug. 2016.
- [11] Dragoljub Gajic, Zeljko Djurovic, et al., "Detection of epileptiform activity in eeg signals based on time-frequency and non-linear analysis," *Frontiers in computational neuroscience*, vol. 9, pp. 38, 2015.
- [12] Juan A. Gallego, Tamar R. Makin, et al., "Going beyond primary motor cortex to improve brain–computer interfaces," *Trends in neurosciences*, 2022.

CONCEPT-BASED EXPLAINABILITY FOR AN EEG TRANSFORMER MODEL

Anders Gjølbye Madsen^{†}*

Áshildur Jónsdóttir^{}*

William Theodor Lehn-Schiøler^{†}*

Bergdís Arnardóttir^{}*

Lars Kai Hansen^{}*

^{*}Technical University of Denmark

Department of Applied Mathematics and Computer Science
2800 Kgs. Lyngby, Denmark

[†]BrainCapture

2800 Kgs. Lyngby, Denmark

ABSTRACT

Deep learning models are complex due to their size, structure, and inherent randomness in training procedures. Additional complexity arises from the selection of datasets and inductive biases. Addressing these challenges for explainability, Kim et al. (2018) introduced Concept Activation Vectors (CAVs), which aim to understand deep models' internal states in terms of human-aligned concepts. These concepts correspond to directions in latent space, identified using linear discriminants. Although this method was first applied to image classification, it was later adapted to other domains, including natural language processing. In this work, we attempt to apply the method to electroencephalogram (EEG) data for explainability in Kostas et al.'s BENDR (2021), a large-scale transformer model. A crucial part of this endeavour involves defining the explanatory concepts and selecting relevant datasets to ground concepts in the latent space. Our focus is on two mechanisms for EEG concept formation: the use of externally labelled EEG datasets, and the application of anatomically defined concepts. The former approach is a straightforward generalization of methods used in image classification, while the latter is novel and specific to EEG. We present evidence that both approaches to concept formation yield valuable insights into the representations learned by deep EEG models.

Index Terms— Explainable AI, EEG Concepts, TCAV, BENDR

1. INTRODUCTION

We investigate representations of electroencephalogram (EEG) data obtained by self-supervised learning methods. Self-supervision is motivated by the lack of labeling in large-scale EEG datasets as labeling is both time-consuming and requires highly specialised EEG expertise. Self-supervised models, such as BERT-inspired Neural Data Representations (BENDR) [1], have the potential to overcome this challenge by learning informative representations from raw, unlabeled data. Such models can subsequently be fine-tuned for downstream classification tasks. We apply the Testing Concept Activation Vectors (TCAV) approach of Kim et al. [2], an interpretability method introduced in 2018, to BENDR-based models, to provide insights into their structure and decision-making processes. See Figure 1 for a conceptual overview. A better understanding of EEG transformer models using TCAV could support the use of these models as diagnostic support tools for identifying EEG abnormalities, such as seizures. However, the question that arises is, what constitutes human-friendly concepts in this context? To address this, we present the following scientific contributions:

This work is supported by The Pioneer Centre for AI, DNRF grant number P1, The Novo Nordisk Foundation grant NNF22OC0076907 "Cognitive spaces - Next generation explainability", and travel grants from The Danish Data Science Academy awarded to AGM and WLS.

- The first TCAV workflows for EEG data, proposing concepts based on human-annotated data as well as concepts defined by human anatomy and EEG frequency ranges.
- Sanity checks for TCAV to ensure valid explanations in simple EEG settings.
- Two practical applications: seizure prediction and brain-computer interfacing.

All code used in this research, along with references to the datasets, have been made publicly accessible for validation and replication¹.

2. THEORY

2.1. BERT-inspired Neural Data Representations

BENDR [1] is inspired by language modeling techniques that have found success also outside text analysis, in self-supervised end-to-end speech recognition and image recognition. It aims to develop EEG models for better brain-computer interface (BCI) classification, diagnosis support, and other EEG-based analyses. Importantly, the approach being based on self-supervision can learn from any EEG data using only unlabeled data. The main goal of BENDR is to create self-supervised representations with minimal robust to context boundaries like datasets and human subjects. The approach is expected to be transferable to future unseen EEG datasets recorded from unseen subjects, different hardware, and different tasks. It can be used as-is or fine-tuned for various downstream EEG classification tasks.

The architecture is based on wav2vec 2.0 [3] developed for speech processing and consists of two stages. The first stage takes raw data, and down-samples it using a stack of short-receptive field 1D convolutions, resulting in a sequence of vectors called BENDR. The second stage uses a transformer encoder [4] to map BENDR to a new sequence related to the target task. Down-sampling is achieved through strides, and the transformer follows the standard implementation with some modifications. The entire sequence is then classified, with a fixed token implemented as the first input for downstream tasks [5]. BENDR differs from the speech-specific architecture in two ways: (1) BENDR is not quantized for pre-training targets, and (2) it has many incoming channels, unlike wav2vec 2.0 which uses quantization and is based on a single channel of raw audio. The 1D convolutions are preserved in BENDR, to reduce complexity. We note that BENDR down-samples at a lower factor than wav2vec 2.0, here resulting in an effective sampling rate of ≈ 2.67 Hz equivalent to a feature window of ≈ 375 ms.

2.2. Linear Head BENDR

For downstream fine-tuning, we use a version where the pre-trained transformer modules are ignored, such that the pre-trained convolu-

¹<https://github.com/AndersGMadsen/TCAV-BENDR>

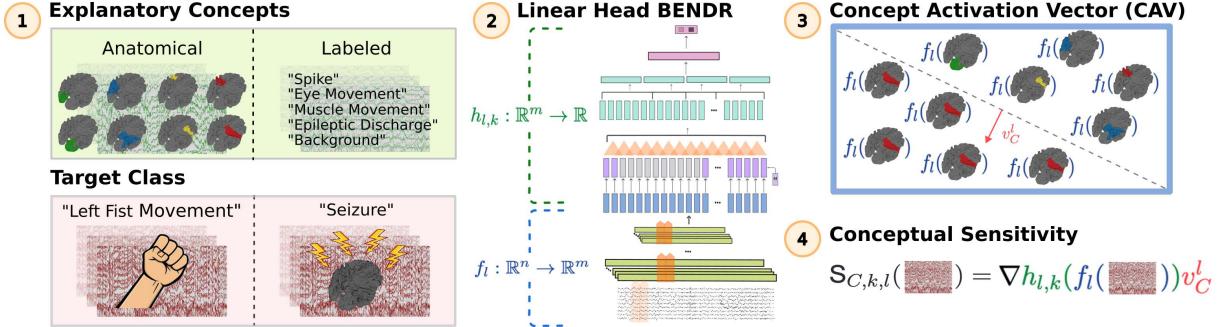


Fig. 1. An overview of using the TCAV method for EEG classification tasks with the Linear Head BENDR model: (1) Explanatory concepts are defined as either event-based EEG labels or frequency-based cortical activity, (2) Layer activations are extracted from a fine-tuned Linear Head BENDR, (3) Concept Activation Vectors (CAV) are defined as the normal vector to the hyperplane separating layer activations for concept data from those of random examples, and (4) The sensitivity of class data for a specific bottleneck of a concept is defined as the directional derivative in the direction of the respective CAV.

tional BENDR stage is used as representation, see [1]. A consistent-length representation is created by dividing the BENDRs into four contiguous sub-sequences, averaging each sub-sequence, and concatenating them. A new linear layer with softmax activation is added to classify the downstream targets based on this concatenated vector of averaged BENDR. We call this the Linear Head BENDR (LHB) model and the structure is illustrated in Figure 2.

The final LHB architecture consists of the following components:

1. **Feature encoder:** Fine-tunes the pre-trained parameters and uses six convolution blocks, each containing a temporal convolution, group normalization, and a GELU activation function to produce a BENDR of length 512.
2. **Encoding augment:** Involves masking and contextualizing the BENDR, with 10% of the BENDR masked and 10% of the channels dropped, while relative positional embeddings from the pre-trained task are added to the BENDR and further preprocessed.
3. **Summarizer:** Applies adaptive average pooling to create four contiguous sub-sequences, averaging each sub-sequence to ensure the model's independence from the input length of EEG recordings.
4. **Extended classifier:** Flattens the four sub-sequences, passes them through a fully connected layer to reduce their dimension, applies a dropout layer, uses a ReLU activation function, and normalizes the output using batch normalization.
5. **Classifier:** Consists of a linear layer with a softmax activation function, which performs the classification task.

2.3. Testing with Concept Activation Vectors (TCAV)

Testing with Concept Activation Vectors (TCAV) is a technique used to quantify the degree to which layers of neural networks align with human-defined concepts [2]. The method is general in the sense that it is not confined to the particular structure of the network nor to the data type. In its essence, TCAV can be broken down into five steps

First, the process involves defining human-aligned concepts and representing them in the data. Alongside these, data from the target class must also be present for evaluation purposes. Furthermore, to establish the directions of the concept activation vector in the latent space, it is necessary to have a collection of concept-negative or random examples.

Second, the layer activations of the concept input and the random input, respectively, are collected and separated by training a binary

linear classifier. Then, the concept activation vector, v_c^l is defined as the normal vector to the hyperplane that separates the two classes (concept vs. random).

Third, for a layer l in the network, the directional derivatives for the target class k along the learned activation vector for concept C is used to calculate how sensitive the prediction of the network is to changes in the input data in the direction of C . We can quantify the sensitivity by

$$S_{C,k,l}(\mathbf{x}) = \nabla h_{l,k}(f_l(\mathbf{x})) \cdot v_C^l, \quad (1)$$

where $h_{l,k}$ is defined as the function that maps activations in layer l through the remaining network and predicts class k .

Fourth, computing the sensitivity for several target examples, $\mathbf{x} \in X_k$, the TCAV score is defined as the ratio of examples that have positive sensitivity, i.e.,

$$\text{TCAV}_{C,k,l} = \frac{|\{\mathbf{x} \in X_k : S_{C,k,l}(\mathbf{x}) > 0\}|}{|X_k|}. \quad (2)$$

In this way, concept activation vectors that are positively aligned with target activations have a TCAV score close to 1 and concept activation vectors that are negatively aligned with target activations have a TCAV score close to 0.

Fifth and final, collecting samples of TCAV scores over several training runs, a suitable statistical test is used to assess the statistical significance of concept activation vectors aligning with the activation of target examples. The null hypothesis of the test is that half of the examples have positive sensitivity and the other half have negative or zero sensitivity, i.e.,

$$H_0 : \text{TCAV}_{C,k,l} = 0.5. \quad (3)$$

Concepts C for which the null hypothesis is rejected thus relate to the target class prediction, and may bring positive or negative evidence for the given target k .

2.4. Source localization

Source localization for EEG data involves mapping electrical signals recorded on the scalp surface to corresponding regions on the cortical surface of the brain. This process uses a head model and the EEG data collected from electrodes placed on the scalp. The reconstruction is a grid of dipolar sources. The solution to this ill-posed problem is called the lead field and there exist many different

Linear Head BENDR

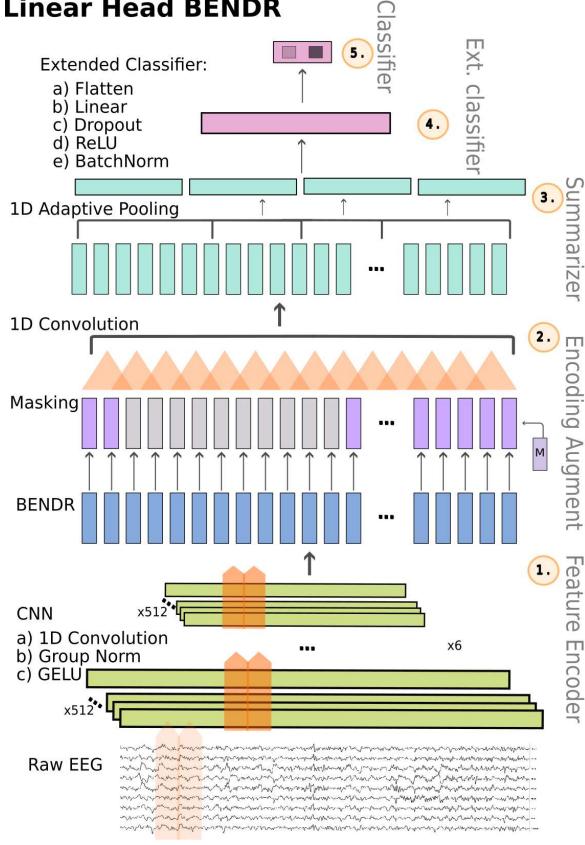


Fig. 2. The Linear Head BENDR (LHB) model architecture illustrated. The model consists of (1) Feature encoder of six confrontational blocks, (2) Encoding augment comprised of masking and convolutional contextualizer, (3) Summarizer using Adaptive Average Pooling, (4) Extended Classifier for dimensionality reduction, and (5) Classifier.

ways to obtain this solution. In this work, we use the exact low-resolution electromagnetic tomography (eLORETA) method implemented in the MNE library [6].

The eLORETA approach presupposes that the EEG measurements of the electric field present on the scalp reflect dipolar sources located in the cerebral cortex. These are conceptually modeled as a three-dimensional distribution of dipoles. The spatial resolution of eLORETA is relatively coarse, which can make pinpointing exact cortical sources challenging. However, for our purpose of estimating aggregated source activity over broadly defined brain regions, such reduced resolution is not an issue.

3. METHODS

3.1. Data

EEG is a non-invasive technique to record the brain's electrical activity. EEG data in this paper refers to these measurements, used often in research and healthcare to identify neurological conditions. In this work, we use five publicly accessible datasets, namely TUH EEG Corpus [7], TUH EEG Artifact (TUAR) Corpus, TUH EEG Events (TUEV) Corpus, TUH EEG Seizure (TUSZ) Corpus [8] and the EEG Motor Movement/Imagery (MMIDB) Dataset [9].

The TUH EEG Corpus contains 69,652 clinical and unlabeled EEG recordings obtained from Temple University Hospital (TUH).

The TUH EEG Artifact Corpus, a labeled subset of the TUH EEG Corpus, includes annotations for five distinct artifacts including eye movement artifact (*eyem*). The TUEV is a subset of the TUH EEG Corpus and includes annotations of event-based EEG segments. There are numerous categories, but we primarily focus on five key classes: (1) technical artifacts (*artf*), (2) background (*bckg*), (3) generalized periodic epileptiform discharge (*gped*), (4) periodic lateralized epileptiform discharge (*pled*), and (5) spike and slow wave (*spsw*). The TUSZ contains EEG signals with manually annotated data for seizure events.

The MMIDB EEG dataset consists of data from 109 participants who are performing or imagining specific motor tasks; our main interest is the moments when subjects either close or imagine closing their left or right fist following a visual cue. We are excluding participants S088, S090, S092, and S100 due to missing data, resulting in 105 participants.

In the construction of brain anatomy concepts, it is imperative to obtain an extensive collection of resting-state EEG data. Due to the limited availability of public datasets with the requisite size and reliability, we utilized The TUH EEG Corpus and source localization to develop a dedicated anatomically labeled resting-state dataset. A set of predefined criteria were employed, including the number of EEG channels, minimum duration, minimum sampling frequency, scaling, and the exclusion of extreme values, which led to the elimination of approximately 90% of the initial EEG recordings. Following this, a manual examination of a part of the remaining data was performed, ultimately yielding 200 human-verified resting-state EEG recordings, corresponding to an aggregate of about 70 hours of EEG data.

In the process of downstream fine-tuning and concept formation, we employ 19 EEG channels, namely *Fp1*, *Fp2*, *F7*, *F3*, *Fz*, *F4*, *F8*, *T7*, *C3*, *Cz*, *C4*, *T8*, *T5*, *P3*, *Pz*, *P4*, *T6*, *O1*, and *O2* (see the MNE documentation [6] for more information). These channels originate from the initial pre-training of BENDR using The TUH EEG Corpus. In instances where the datasets lack these channels, we establish the following mapping: $T3 \mapsto T7$, $T4 \mapsto T8$, $P7 \mapsto T5$, and $P8 \mapsto T6$. We also resample the corresponding EEG data to a 256 Hz sampling frequency and apply a high-pass FIRWIN filter with a 0.1 Hz cutoff, a low-pass FIRWIN filter with a 100.0 Hz cutoff, and a 60 Hz FIRWIN notch filter to eliminate powerline noise. In situations where preprocessing cannot be performed, the EEG is excluded. Finally, we scale each trial to the range $[-1, 1]$ and append a relative amplitude channel, see [1], resulting in a total of 20 channels.

3.2. Training

Pre-training of BENDR is based on the large set of unlabelled EEG data from The TUH EEG Corpus. The pre-training procedure is largely based on wav2vec 2.0 and involves two main stages: The convolutional stage and the transformer stage. The convolutional stage generates a sequence of representations (BENDRs) that summarize the original input. This sequence is then fed into the transformer stage, which adjusts its output to be most similar to the encoded representation at each position. The layers affected during pre-training are the feature encoder and the transformer. Kostas et al. [1] kindly made the pre-trained weights of the encoder and contextualizer publicly available, and this is the model that we have employed here.

The LHB model architecture described in Figure 2 is used for downstream fine-tuning. We aim to optimize the model for two distinct binary classification objectives. First, the model is fine-tuned for the differentiation between *seizure* and *non-seizure* events, using

the TUSZ Corpus with 60-second window segments. The hyperparameters are determined using Bayesian optimization to maximize the validation F_1 -score. The fine-tuning employs a batch size of 80, a learning rate of 1×10^{-4} , and 30 epochs. This results in a model with a balanced accuracy of 0.73 ± 0.07 .

In our second fine-tuning example, the model is adapted for the differentiation between *Left Fist Movement* versus *Right Fist Movement*, using the MMIDB EEG Dataset with 4-second window segments. We are using both the imaginary and performed task data from the 105 participants. We train the model for 7 epochs with a batch size of 4 and a learning rate of 1×10^{-5} . The hyperparameters were chosen based on the best validation balanced accuracy from leave-one-subject-out cross-validation where the model was trained for 50 epochs and the best model was retained. The specific hyperparameter configuration aligns with the optimal hyperparameters found by the original authors [1] and we find a similar balanced accuracy of 0.83 ± 0.02 .

3.3. Constructing Concepts

To construct human-aligned explanatory EEG concepts, a number of initial investigations were conducted. The data processing involved follows the methodology previously mentioned. In this section, we provide a general pipeline overview and discuss several choices made throughout the process.

Concepts from Labeled EEG Data: Using the labeled EEG data from the TUAR and TUEV Corpus and the MMIDB EEG Dataset, we create concepts representing activities within specific time windows. Each annotated segment of the EEG data is divided into windows of predetermined length and assigned the corresponding label.

In the TUEV Corpus, we define concepts for the spike/short wave (*spsw*), periodic lateralized epileptic discharge (*pled*), general period epileptic discharge (*gped*), technical artifact (*artf*), and background (*bckg*) with 60-second windows. This approach aligns with the length of the *seizure* classifier.

Lastly, we examine the eye movement (*eyem*) from the TUAR Corpus and *Left Fist Movement* and *Right Fist Movement* from the MMIDB EEG Dataset, both using 4-second windows. These different-sized windows then constitute examples of concepts defined based on their labels.

Anatomical Concepts from Unlabeled EEG Data: The objective is to identify concepts representing specific frequency bands within distinct areas of the cortex, e.g. *alpha activity in pre-motor cortex* or *gamma activity in early visual cortex*. To obtain a non-task-specific representation of each cortical area, we utilize resting-state EEG data, as it spontaneously generates activity throughout the cortex. For this purpose, we use a subset of The TUH EEG Corpus, as described above.

To define anatomical concepts, EEG data is segmented into 4-second windows, with the first and last 5 seconds of each sequence excluded to minimize artifact contamination. The data is then divided into five frequency bands with a FIRWIN bandpass filter: *delta* (0.5-4Hz), *theta* (4-8Hz), *alpha* (8-12Hz), *beta* (12-30Hz), and *gamma* (30-70Hz). The inverse operator for the forward model is computed using eLORETA [6] via the MNE Python library. Since the spatial resolution is not critical, minimal regularization of 1×10^{-4} is applied.

Using the combined version of the multi-modal parcellation of the human cerebral cortex, HCPMMP1 [10] and the inverse operator, the average power of electrical activity in 23 cortical areas for each hemisphere is determined.

Our interest lies in cortical areas exhibiting the greatest deviation

from typical activity within a specific frequency band. However, cortical areas are not equidistant from the scalp or consistent in baseline activity across bands. To normalize for these differences in the distribution of cortical activity, we compute the mean and standard deviation of the power in each cortical area for each frequency band on an EEG session level, which will be employed in various ways. We call these the baseline mean and the baseline standard deviation.

We explore possible approaches to how the baseline means and standard deviation for each EEG session could be used to normalize the power of 4-second windows within that session. The options include dividing by the baseline standard deviation to account for scalp source variation, subtracting or dividing by the baseline mean to identify the cortical area with the greatest deviation, taking the absolute difference or not, and selecting a single cortical area across all frequency bands or only within a specific band.

Identifying a single frequency and cortical area for each 4-second window of EEG data is a challenging task without prior work to guide the process, and each method presents its own limitations. We specifically look for *alpha* desynchronization in the cerebral cortex during imagined or actual movement and closed or open eyes in the MMIDB EEG dataset, i.e., that *alpha* activity in cortical areas decreases when activated. Using a paired t-test to examine the presence of lateralization in cortical activities for different methods, we found that the preferred approach is to choose the area which maximizes the absolute difference between the given time window's power and the baseline mean, divided by the baseline standard deviation, only within specific frequency bands.

Random Concepts: Construction of CAVs calls for data examples that are considered random with respect to the concept of interest. In all experiments, random concepts consisting of 4-second or 60-second windows were randomly sampled from resting-state data obtained from the subset of the TUH EEG Corpus and unannotated sections of the TUAR dataset.

3.4. Experiments

We investigate two approaches for defining explanatory concepts in EEG data. The TCAV method is then employed to evaluate whether the LHB model uses specifically defined human-aligned concepts of EEG data. For all concepts, the resulting activation vectors for all five bottlenecks in the LHB model architecture are examined to determine if they significantly align with the latent representations of class data in the model. We conduct the following experiments:

1. **Sanity Checks:** We verify the TCAV method and construction of concepts function as intended through a series of sanity checks when classifying *Left Fist Movement*.
2. **Event-based Concepts:** We assess whether the LHB model leverages specific EEG events in the classification of *seizure*.
3. **Anatomy/Frequency-based Concepts:** We investigate if the LHB model employs lateralization in cortical activity in the *alpha* band for classifying *Left Fist Movement*. The chosen cortical areas are based on their relevance to the classification task.

In the experiments, we use the TCAV method with a regularized linear model and stochastic gradient descent (SGD) learning, setting the regularization parameter $\alpha = 0.1$ to learn the decision boundary between explanatory and random concepts. We employ 50 random concepts and a maximum of 40 examples per concept. These parameters were chosen to increase statistical power. The mean TCAV scores for the target concept examples and the random examples are compared using the non-parametric Mann-Whitney U Rank test, as

opposed to the t-test used in the original TCAV method, as we observed a clear violation of the normality assumption for the TCAV scores. To mitigate Type I errors, the p-values are corrected for each experiment employing the conservative Bonferroni method, after which we claim significance if the corrected p-value is below 0.05.

4. RESULTS

4.1. Sanity Checks

We first provide evidence that the TCAV method can be applied to explain EEG data and the LHB model. In Figure 3, the high significance of class data as concepts (*Left Fist Movement* with positive evidence and *Right Fist Movement* with negative evidence) confirms this. Furthermore, concepts based on maximal activity in either the left or right hemisphere for the *alpha* frequency band strongly indicate that lateralized cortical activity is detected by several layers in the model, as expected.

Moreover, the negative alignment of a concept based on labeled artifacts with the model representation of motor task data implies that artifacts in EEG data significantly influence classification tasks. We find that *eyem* has a negative impact on the classification of *Left Fist Movement*. Note that this does *not* mean that *eyem* positively affects the opposite class, that is *Right Fist Movement*, as the TCAV Score is specific to the "*Left Fist Movement* dataset". Conversely, *eyem* could negatively affect the classification of both *Left Fist Movement* and *Right Fist Movement*, due to the lower signal-to-noise ratio for classification when artifacts are present.

4.2. Event-based concepts

We next investigate whether fine-tuning the LHB model for seizure classification on the TUSZ dataset and using explanatory concepts defined with labeled data from TUEV aligns with the model's internal representation for data labeled as containing seizures. The target of the investigation is the *seizure* label and we test all bottlenecks in the LHB model. The results of this experiment are shown in Figure 4.

When compared to EEG data labeled as containing seizures, the epilepsy-related concepts *pled*, which is present in certain brain areas, and *gped*, which is present in most of the brain, exhibit high and positive evidence in nearly all bottlenecks. This observation

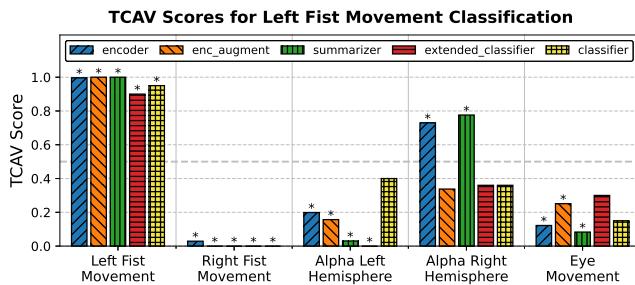


Fig. 3. Sanity checks for applying the TCAV method to EEG data and the bottlenecks of the LHB model. The figure presents the results of TCAV for the *Left Fist Movement* class in a binary classification task using the MMIDB EEG dataset. From right to left, concepts are defined as follows: (1) *Left Fist Movement* and (2) *Right Fist Movement* class data, maximal mean activity in the alpha frequency band for (3) *Left Hemisphere* and (4) *Right Hemisphere*, respectively, and (5) *Eye Movement* artifacts. Stars indicate either positive (a score above 0.5) or negative (a score below 0.5) statistical significance.

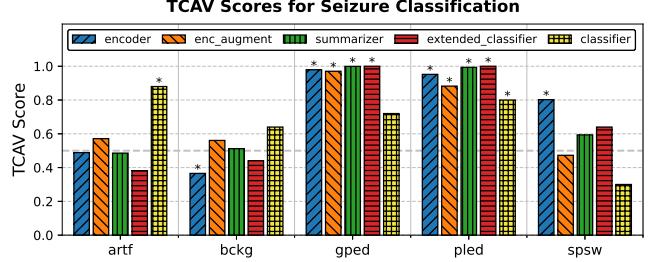


Fig. 4. The results of utilizing TCAV to assess whether event-based EEG labels align with the internal representation of the *seizure* class data in the LHB model at the five bottlenecks are presented. From the right, the concepts are defined as (1) technical artifacts (*artf*), (2) background (*bckg*), (3) generalized periodic epileptic discharge (*gped*), (4) periodic lateralized epileptic discharge (*pled*), and (5) spike and short wave (*spsw*). Stars indicate either positive (a score above 0.5) or negative (a score below 0.5) statistical significance.

aligns with existing literature that associates epileptiform discharges with seizures [11], and it is expected that the LHB model will use these properties for classification. The *spsw* concept also demonstrates significant positive evidence in the *encoder* bottleneck but not in the further downstream bottlenecks. Similarly, the *bckg* concept shows negative evidence in the *encoder* bottleneck but not in the further downstream bottlenecks. It is interesting that these concepts only come to be significant in the initial bottleneck. A possible explanation is that the technical artifacts *artf* and *bckg* are not significant for the classification, but BENDR effectively identifies seizure-related concepts and filters out noise. The results also suggest that the model's *classifier* and *extended classifier* can be further optimized, as *artf* is near-significant level in these bottlenecks and, as a result, the noise has not been completely removed. In conclusion, these examples indicate that concept-based explainability can provide valuable model design information.

4.3. Anatomy/Frequency-Based Concepts

We have demonstrated that labeled EEG data can generate human-aligned concepts, which are integrated into the LHB model for seizure classification. This comes quite naturally as labeled data is labeled by humans and tend to align with human-relatable concepts. We then present evidence that defining explanatory concepts based on cortical activity in frequency bands may uncover patterns corresponding to the model's internal representations.

In particular, for a motor classification task using the MMIDB EEG dataset and targeting the *Left Fist Movement* class, we show that cortical activity in the *alpha* band aligns with the model's internal representation. In Figure 5, we find that the CAV for *Somatosensory and Motor Cortex* in the right hemisphere positively aligns with the activations of *Left Fist Movement* class data across all bottlenecks in the model. The mean TCAV scores are also consistently positively significant. At the same time, the TCAV scores for the same cortical area in the *Left Hemisphere* are either negatively significant or insignificant. These results strongly suggest that the model's internal representation incorporates lateralization, reflecting the fact that one hemisphere exhibits more electrical activity than the other. It is noteworthy that lateralization is most significant in the *Encoding Augment* and *Summarizer* bottlenecks, indicating that it is captured early in the network.

Additionally, we observe that the *Primary Visual Cortex (V1)* areas do not exhibit lateralization, and their TCAV scores are insignificant across all bottlenecks and for both hemispheres. This further

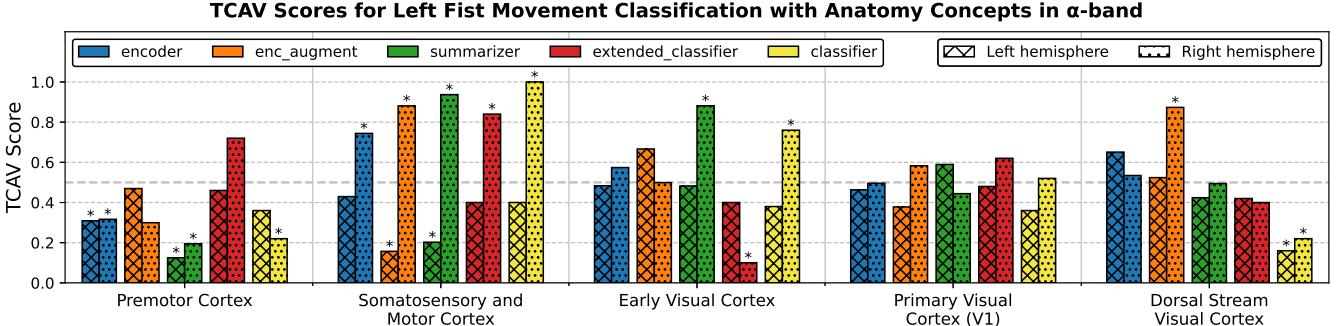


Fig. 5. Using TCAV, we analyzed the alignment between anatomical concepts in the *alpha* band and the internal representation of the *Left Fist Movement* class in the LHB model at five bottlenecks. The visualization of five pairs of concepts focused on five cortical areas, located in both the left and right hemispheres, that were deemed most relevant for the classification task. The chosen concepts had a higher deviation in the *alpha* band. Stars indicate either positive (a score above 0.5) or negative (a score below 0.5) statistical significance. Our analysis reveals significant lateralization in the *Somatosensory and Motor Cortex* across all five bottlenecks. Additionally, we observe that the *Primary Visual Cortex (VI)* was insignificant for both hemispheres in all bottlenecks.

supports the conclusion that the LHB model utilizes specific cortical areas in its classification rather than all areas indiscriminately.

While no apparent lateralization is present in the *Premotor Cortex*, this part of the cortex is negatively significant in the *Encoder* and *Summarizer* bottlenecks for both the left and right hemispheres. A possible explanation is that the instances we examine involve participants *performing* movements; therefore, there may not necessarily be relevant activity in the *Premotor Cortex*, which is primarily involved in movement planning [12].

Lastly, we observe significance in the *Classifier* bottleneck for *Early Visual Cortex* and *Dorsal Stream Visual Cortex*. We note that the movement is activated by a visual cue; however, further experiments would be required to fully clarify the effect.

5. CONCLUSION

Concept-based explainability has proven to be valuable in various domains, such as image classification and natural language understanding, where concepts are naturally defined using labeled data. In this study, we have explored the definition of concepts for EEG models for the first time. We presented two new workflows for concept-based explainability within the TCAV framework for EEG data. First, we adopted an approach akin to the original work of Kim et al. [2], in which concepts are derived from labeled data. In this case, we utilized various annotated EEG databases, e.g., data from the Temple University Hospital EEG database. The second workflow is based on the source location of resting-state EEG data also from the Temple University Hospital database. This enables us to generate datasets for TCAV derived from anatomical brain areas and for specific frequency bands, e.g., the *alpha* band. We demonstrated a proof of concept through several "sanity check" experiments to verify expected responses in elementary EEG settings, such as EEG lateralization during left- or right-hand movement. Lastly, we examined two practical applications: A case study involving seizure prediction, where TCAV reveals the role of fundamental spike patterns, and a brain-computer interface case, hinting at how the TCAV method can assist in debugging and offer valuable insights into classifier design for EEG data.

6. REFERENCES

- [1] Demetres Kostas, Stéphane Aroca-Ouellette, et al., "Bendr: Using transformers and a contrastive self-supervised learning

task to learn from massive amounts of eeg data," *Frontiers in Human Neuroscience*, vol. 15, 2021.

- [2] Been Kim, Martin Wattenberg, et al., "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav)," 2018.
- [3] Alexei Baevski, Henry Zhou, et al., "wav2vec 2.0: A framework for self-supervised learning of speech representations," *CoRR*, vol. abs/2006.11477, 2020.
- [4] Ashish Vaswani, Noam Shazeer, et al., "Attention is all you need," 2017.
- [5] Jacob Devlin, Ming-Wei Chang, et al., "BERT: Pre-training of deep bidirectional transformers for language understanding," Minneapolis, Minnesota, June 2019, pp. 4171–4186, Association for Computational Linguistics.
- [6] Roberto D. Pascual-Marqui, Dietrich Lehmann, et al., "Assessing interactions in the brain with exact low-resolution electromagnetic tomography," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 369, no. 1952, pp. 3768–3784, 2011.
- [7] Amir Harati, Silvia Lopez, et al., "The tuh eeg corpus: A big data resource for automated eeg interpretation," 12 2014.
- [8] Vinit Shah, Eva von Weltin, et al., "The temple university hospital seizure detection corpus," *Frontiers in Neuroinformatics*, vol. 12, 2018.
- [9] Gerwin Schalk, Dennis J. McFarland, et al., "Bci2000: a general-purpose brain-computer interface (bci) system," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 1034–1043, 2004.
- [10] Matthew F. Glasser, Timothy S. Coalson, et al., "A multi-modal parcellation of human cerebral cortex," *Nature*, vol. 536, no. 7615, pp. 171–178, Aug. 2016.
- [11] Dragoljub Gajic, Zeljko Djurovic, et al., "Detection of epileptiform activity in eeg signals based on time-frequency and non-linear analysis," *Frontiers in computational neuroscience*, vol. 9, pp. 38, 2015.
- [12] Juan A. Gallego, Tamar R. Makin, et al., "Going beyond primary motor cortex to improve brain–computer interfaces," *Trends in neurosciences*, 2022.

LARGE TRANSFORMERS ARE BETTER EEG LEARNERS

Bingxin WANG¹, Xiaowen Fu¹, Yuan LAN¹, Luchan ZHANG^{3*}, Wei ZHENG^{4*}, and Yang Xiang^{1,2*}

¹Department of Mathematics, The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong SAR, China
²Algorithms of Machine Learning and Autonomous Driving Research Lab, HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute, Futian, Shenzhen, China

³College of Mathematics and Statistics, Shenzhen University, Shenzhen 518060, China

⁴Shenzhen Youjia Innov Tech Co., Ltd, Shenzhen, China

*Corresponding authors: zhengwei@mineye.cc, zhanglc@szu.edu.cn, maxiang@ust.hk

ABSTRACT

Pre-trained large transformer models have achieved remarkable performance in the fields of natural language processing and computer vision. However, the limited availability of public electroencephalogram (EEG) data presents a unique challenge for extending the success of these models to EEG-based tasks. To address this gap, we propose AdaCT, plug-and-play **Adapters** designed for **Converting Time series data into spatio-temporal 2D pseudo-images or text forms**. Essentially, AdaCT-I transforms multi-channel or lengthy single-channel time series data into spatio-temporal 2D pseudo-images for fine-tuning pre-trained vision transformers, while AdaCT-T converts short single-channel data into text for fine-tuning pre-trained language transformers. The proposed approach allows for seamless integration of pre-trained vision models and language models in time series decoding tasks, particularly in EEG data analysis. Experimental results on diverse benchmark datasets, including Epileptic Seizure Recognition, Sleep-EDF, and UCI HAR, demonstrate the superiority of AdaCT over baseline methods. Overall, we provide a promising transfer learning framework for leveraging the capabilities of pre-trained vision and language models in EEG-based tasks, thereby advancing the field of time series decoding and enhancing interpretability in EEG data analysis. Our code will be available at <https://github.com/wangbxj1234/AdaCE>.

1 Introduction

Electroencephalography (EEG) has long been instrumental in unraveling the intricacies of the human brain. In contrast to text, audio, or video, modeling EEG data presents three main challenging characteristics. Firstly, the magnitude of available public electroencephalogram (EEG) data is significantly lower than that of text and image data. This poses a unique obstacle for transformer models pre-trained from EEG data, making it nearly impossible to reach the scale of other large transformers. Secondly, multi-channel EEG data analysis for predictive and classificatory purposes requires a comprehensive understanding of both its temporal and spatial dimensions. The temporal aspect refers to the evolution of neuroelectric signals over time, while the spatial dimension involves the distribution of these signals across multiple channels. Extracting meaningful information from EEG data relies on interpreting its inherent texture, which is manifested in the time-spatial dependency derived from multi-channel recordings. Thirdly, in most existing EEG prediction or classification tasks, the primary objective is to capture immediate stimuli or distinctive pre-ailment patterns from lengthy time series data. However, a significant challenge arises due to the inherent quadratic complexity of attention computation, imposing limitations on the input length of transformer-based models. This presents a hurdle in transforming the raw EEG data, with its extended sequences, into a format suitable for the model without compromising information integrity.

To bridge the gap between the scarcity of available EEG data and the potential of large transformer models pretrained on other modalities, in this paper, we demonstrate that large models (LMs) pre-trained from images as well as text can be fine-tuned for EEG-based prediction tasks without introducing extra parameters to be trained. In particular, we introduce AdaCT, plug-and-play **Adapters** designed for converting time series data into spatio-temporal 2D pseudo-images or text, to apply pretrained LMs for EEG prediction problems, with the framework illustrated in Fig. 1. Fundamentally,

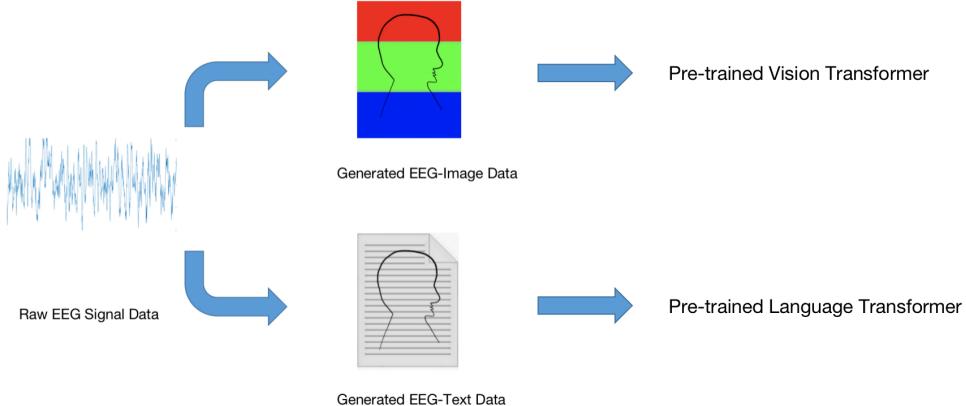


Figure 1: Framework: Adapters for converting time series EEG data into images or text for fine-tuning pre-trained large transformers.

this method converts multi-channel or lengthy single-channel time series data into spatio-temporal 2D pseudo-images, while converting short single-channel data into text.

To achieve robust performance, we propose a three-stage method for converting EEG data into pseudo-images compatible with pre-trained visual transformers while maximizing the retention of pertinent information. Leveraging the enhanced generalization capabilities acquired through pre-training, we show how pre-trained vision transformers can capture the complete texture features embedded in the spatio-temporal EEG pseudo-images. For adapting short single-channel EEG to pre-trained language transformer models, we convert the temporal dynamics of the EEG signal into a text-based representation, leveraging the inherent sequential nature of both the EEG data and text.

AdaCT seamlessly integrates with pre-trained transformers, facilitating the application of cutting-edge models to EEG prediction without introducing extra parameters. Our evaluations reveal AdaCT’s ability to surpass or match state-of-the-art EEG prediction methods. Furthermore, it achieves top-tier results on other multi-channel human activity recording time series datasets, including the UCI HAR benchmark[1].

2 Related Works

2.1 Transformer-Based EEG Decoding Methods

In the area of EEG decoding, transformer-based models have been leveraged for enhancing the capture of long-term dependencies in studies such as those conducted by [2], [3], and [4]. Each of these investigations significantly contributed to advancing this research paradigm. A common methodology across these studies involved the incorporation of a combined Convolutional Neural Network (CNN) module. This module, executing convolutions separately across the dimensions of time and space, facilitated the pre-extraction of both temporal and spatial features. These approaches collectively represent a nuanced strategy for empowering models to discern intricate patterns in EEG data, thereby paving the way for advancements in the understanding and decoding of complex neural signals.

However, introducing a CNN pre-extraction module to pre-trained large transformers can pose challenges. This addition introduces additional parameters that require training. During the fine-tuning process, backpropagating the gradients of these parameters throughout the entire model significantly increases computational costs.

2.2 BERT-like Transformers Pre-trained on EEG Datasets

The pioneering studies conducted by [5] and [6] have set the stage for a transformative approach in EEG decoding. These investigations stand out for their ingenious utilization of pre-trained BERT-like transformers, marking a commendable leap forward in the convergence of natural language processing and neuroscientific research. Leveraging pre-trained language models to decode EEG data represents a novel and promising avenue, showcasing the adaptability and potential of transformer technology in a field traditionally dominated by more conventional methods.

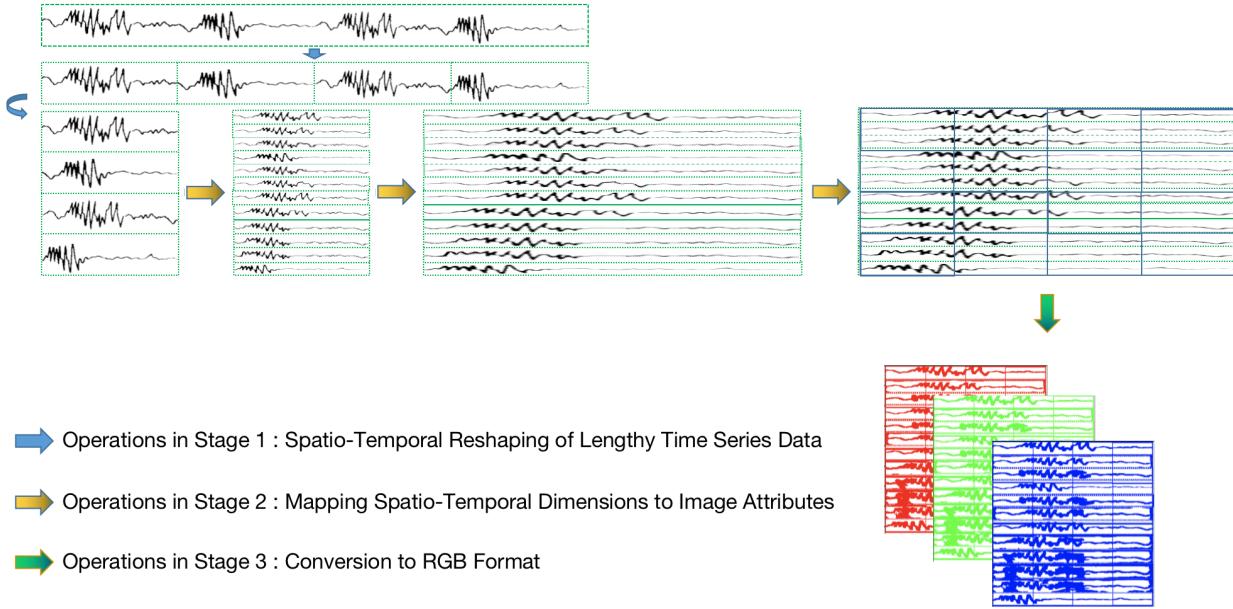


Figure 2: Illustration of the AdaCT-I method, showcasing the spatio-temporal reshaping, mapping to image attributes, and conversion to RGB format steps for converting time series data into two-dimensional RGB images.

However, this innovative approach encounters inherent challenges. The scalability of pre-trained transformers on EEG datasets is constrained by the limited availability of public EEG data compared to more abundant text and image datasets. Furthermore, the application of large language transformers faces hurdles related to managing input sequence length. The quadratic complexity of attention calculation imposes limitations on the allowable length of input sequences, with challenges arising particularly when attempting to consolidate multiple channels into a unified text input.

2.3 Generative Pre-trained Transformer (GPT)

Generative Pre-trained Transformer (GPT) [7] models, pre-trained on vast textual corpora, have garnered acclaim in natural language processing tasks. Their effectiveness lies in their ability to grasp contextual dependencies and generate coherent text. This pre-training enables GPT to capture intricate linguistic patterns and nuances, making it a powerful tool for language-related applications. The adaptation of pre-trained GPT models in fields beyond language processing is an ongoing exploration. Our proposed transfer learning framework contributes to this exploration by facilitating their application in EEG decoding tasks, where the pre-trained models demonstrate promising generalization capabilities.

2.4 Pre-trained Vision Transformer (ViT)

Vision Transformer, introduced by [8], revolutionized computer vision by applying transformer architectures directly to image data. Unlike traditional convolutional neural networks (CNNs), ViT abandons convolutions in favor of self-attention mechanisms. The image is divided into fixed-size patches, linearly embedded, and processed by transformer layers, enabling the model to capture both local and global features. By pre-training on extensive image datasets, ViT learns rich visual representations and global dependencies. This pre-trained knowledge, when fine-tuned for specific visual tasks, accelerates convergence and demonstrates superior performance in discerning local and global patterns. Based on the insights discussed above, we believe that once a reasonable method is found to convert the time series data into pseudo-images, the pre-trained ViT model can serve as an effective feature extractor for EEG classification tasks.

3 AdaCT: Adapters for Converting Time Series Data into Images or Text

In this section, we delineate our solution crafted for fine-tuning pre-trained transformers on EEG datasets. To enhance its applicability, we have developed plug-and-play adapters designed for transforming raw EEG data into a format directly compatible with general pre-trained transformer models. The purpose of this approach is to leverage the

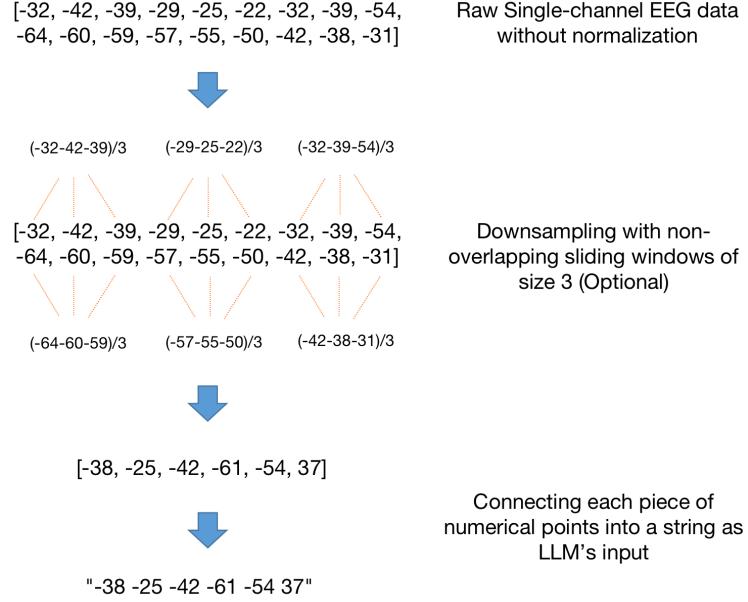


Figure 3: Illustration of the AdaCT-T method, highlighting the non-overlapping sliding window downsampling step for converting time series data into text representation.

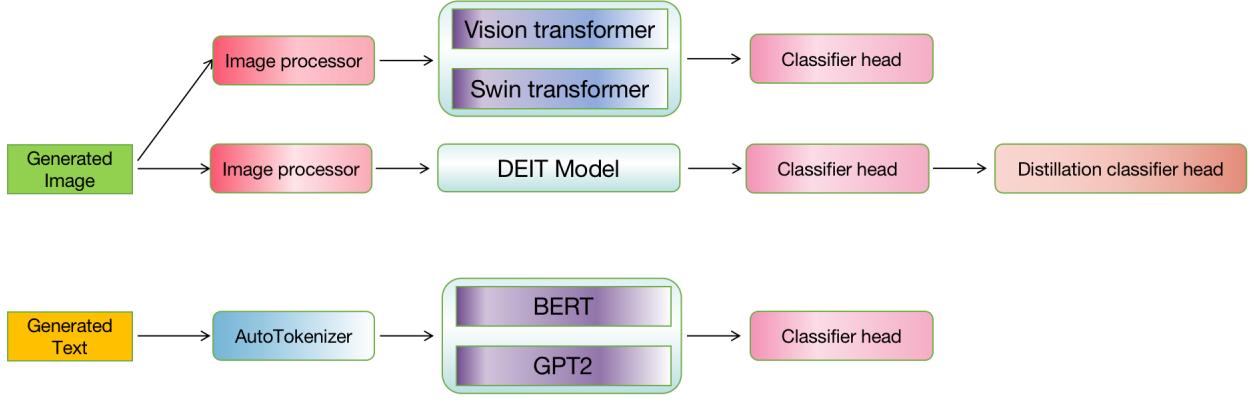


Figure 4: Overview of the fine-tuning process for pre-trained vision transformers and language transformers on converted EEG datasets. The process involves image processing for vision transformers and tokenization for language transformers, followed by integration with pre-trained models and classification head modules.

generalization capabilities inherent in transformer models pre-trained on diverse datasets to improve the performance of EEG prediction and classification tasks.

Generally, with the aim of more effectively harnessing the diverse characteristics of pre-trained visual and language transformers, we seek to partition distinct transformation approaches based on the length and channel count of the data. Subsequently, the following sections intricately expound upon the specific implementations of the proposed adapters, namely AdaCT-I and AdaCT-T. AdaCT-I concentrates on the conversion of multi-channel or lengthy EEG data into spatio-temporal 2D pseudo-images. Simultaneously, AdaCT-T investigates the adjustment of pre-trained language transformers for short single-channel EEG, translating temporal dynamics into a text-based representation.

3.1 AdaCT-I: Adapt Time Series Data into Images

Our goal is to convert raw EEG data into two-dimensional RGB images, emphasizing the preservation of complete texture characteristics. The adaptation process includes three stages, as shown in Fig. 2.

3.1.1 Spatio-Temporal Reshaping of Lengthy Time Series Data

Handling lengthy time series introduces challenges, such as potential information loss and model input constraints. In addressing these challenges, we found that the core of most EEG prediction or classification tasks is to capture immediate stimuli or distinctive pre-ailment patterns, in which the contribution of long temporal dependencies is insignificant. Based on this insight, we propose partitioning the signal into evenly spaced patches along the temporal axis and stacking them spatially, augmenting the spatial dimensionality. Instead of focusing on preserving extended temporal correlations, our approach prioritizes essential features for effective modeling. For instance, in the case of single-channel EEG data with a time series length of 65536, we treat every 256 points as a patch, reshaping it into 256x256 spatio-temporal data. This pre-adjustment ensures the model effectively captures critical information without being burdened by unnecessary temporal intricacies, especially in scenarios with limited channels and extended time durations.

3.1.2 Mapping Spatio-Temporal Dimensions to Image Attributes

Transforming multi-channel time series EEG data into a 2D pseudo-image centers on establishing a direct mapping from the sampling points to image pixels. In our approach, each sampling point in the time series corresponds to a pixel, and linear interpolation is applied to emphasize channel data importance and enhance inter-channel correlation. This ensures the preservation of essential spatio-temporal features critical for subsequent tasks.

3.1.3 Conversion to RGB Format

In the final stage, we convert the 2D pseudo-images into RGB format, ensuring seamless compatibility with the input interfaces of most pre-trained visual transformers. Notably, vision transformers will flatten the RGB channels of input images before the attention calculation, for example, flattening (16, 16, 3) into a vector of size (16*16*3). To preserve the original structure and content, we apply the inverse operation of flattening, namely folding, to convert the 2D pseudo-images into RGB format. This folding operation allows us to reconstruct the initial 2D pseudo-images from the flattened representation without any kind of information loss.

3.2 AdaCT-T: Adapt Time Series Data into Text

To adapt pre-trained transformer models to single-channel EEG data, we propose a straightforward yet effective approach. We convert the temporal dynamics of the EEG signal into a text-based representation, leveraging the inherent sequential nature of both the EEG data and text. This involves intuitively connecting sequential sampling points with spaces, creating a coherent text string. Prior to implementing this mapping, we conduct a two-stage process to optimize the input format for subsequent model utilization.

3.2.1 Temporal Data Scaling for Signal Representation

For a given dataset, following an examination of the overall data distribution, we employ a consistent scaling method to convert each temporal data point of the EEG signal into a three-digit integer. It is important to note that, although datasets may showcase diverse distributions, data within the same dataset typically adheres to consistent upper and lower limits. Thus, to guarantee a succinct numeric representation for each temporal data point, we adapt scaling methods based on the unique characteristics of each dataset. As an illustration, for a normalized dataset, we utilize a scaling approach that involves multiplying by a thousand and subsequently rounding the result.

3.2.2 Non-overlapping Sliding Window Downsampling

When addressing EEG data, the densely sampled nature introduces pronounced continuity characteristics among neighboring sampling points on the timestamp. Exploiting this feature, we apply a non-overlapping sliding window downsampling technique, adjusting the data to meet the maximum input length of the target language transformer, as illustrated in Fig. 3.

The crux of this approach lies in employing non-overlapping sliding windows: within each window, the average of sampling points is computed for downsampling. It is crucial to emphasize that we deliberately choose a relatively small

window size (typically set to 3 in most experiments). This strategic choice proves beneficial as it condenses extensive time series data while preserving fundamental temporal texture information.

3.3 Fine-tune Pre-trained Vision Transformers on Converted Datasets

Following the transformation of EEG, we align the generated image and text datasets with the transformer model’s input requirements for their utilization in fine-tuning pre-trained transformers. For AdaCT-I, the pre-folded spatio-temporal dimensions of the 2D pseudo-images undergo standard patch cutting processes, dividing them into smaller patches for the transformer to efficiently process localized information. Concurrently, in AdaCT-T, tokenization is applied to the text-based representation of EEG data. The sequential nature of the text format aligns seamlessly with the transformer’s capacity to capture dependencies among tokens, facilitating effective training. The overall framework is depicted in Fig. 4.

Fine-tuning pre-trained vision transformer models for EEG data processing involves a sequence of steps. We provide a detailed explanation of the architecture in three modules:

3.3.1 Image Processing Module

The objective of the image processing module is to prepare the pseudo-images of EEG data for compatibility with pre-trained vision transformer models. This transformation is accomplished through the following operations:

- **Resizing and Standardization:** The image processor firstly resizes the input images to a fixed size, ensuring uniformity and compatibility with the requirements of the vision transformer models. Additionally, color channels are standardized, and pixel values are normalized to promote uniformity across different images.
- **Normalization:** Following resizing, normalization techniques are applied to standardize the pixel values of the input images to a predefined scale, typically ranging from 0 to 1 or -1 to 1. This normalization step stabilizes the training process and enhances the convergence of vision transformer models during subsequent fine-tuning.
- **Augmentation and Encoding:** Subsequently, augmentation techniques are employed to enhance the diversity of the converted training data artificially. Techniques such as rotation, flipping, and cropping contribute to improving the robustness and generalization capabilities of the vision transformer models. Finally, the preprocessed images are encoded into numerical tensors or vectors, preserving both spatial and semantic information, thus rendering them suitable for input into the vision transformer models.

The image processing module ensures the effective transformation of input images into a standardized format encapsulating the essential visual features and semantics of EEG data. This standardized representation facilitates subsequent processing by transformer models, enabling efficient extraction of meaningful insights from the input data. Ultimately, these enhancements contribute to the improved performance and accuracy of EEG-related tasks such as prediction and classification.

3.3.2 Integration with Pre-trained Vision Transformer Models

After being processed by the image processor module, the preprocessed images are primed for direct integration into cutting-edge vision transformer models to extract features. In this paper, we select three popular vision transformer architectures to serve as feature extractors for the pseudo-images generated by AdaCT-I: ViT [8], Swin Transformer [9], and DeiT [10]. The integration of preprocessed EEG pseudo-images with vision transformer models is crucial to ensuring high-precision prediction. In the following discussion, we analyze the characteristics of three vision models and their suitability for processing EEG-derived data.

- **Vision Transformer (ViT):** In the architecture of ViT [8], the input image is divided into fixed-size patches, linearly embedded, and processed by transformer layers. Moreover, it is often observed that immediate stimuli or distinctive pre-ailment patterns are represented by only a small subset of adjacent original sampling points. Therefore, through the spatio-temporal reshaping of EEG data by AdaCT-I and the patch-wise partitioning approach of ViT, most of the critical texture information remains intact within each patch. By capitalizing on its inherent ability to capture spatial and temporal patterns effectively, ViT emerges as a robust choice for our task. Its architecture is adept at handling the nuanced complexities inherent in the transformed EEG data, thus facilitating the extraction of meaningful features for subsequent analysis.
- **Swin Transformer:** The Swin Transformer [9] incorporates shifted windows, enabling efficient capture of correlated features across different windows compared to ViT. While ViT generally retains most waveforms containing key EEG texture information within the same patch, in some instances, the required texture is

dispersed across multiple patches after partitioning. The window shifting technique effectively addresses this issue, mitigating information loss and facilitating comprehensive feature extraction across various spatial contexts. Moreover, the Swin Transformer’s hierarchical architecture facilitates feature aggregation across multiple scales, allowing it to extract both local and global information from the input data. These characteristics enable the Swin Transformer model to extract features effectively from the EEG-derived data.

- **Data-efficient Image Transformer (DeiT):** The DeiT model [10] is a distilled Vision Transformer that utilizes a distillation token to learn from a CNN teacher during pre-training on conventional image datasets. This approach contributes to enhanced generalization of the model, enabling it to transfer its image feature extraction capabilities to datasets generated by AdaCT-I. Additionally, DeiT’s architecture is specifically designed to efficiently process image data and capture both local and global features, making it well-suited for learning the inherent texture with the transformed EEG images.

The vision models we used were pre-trained on ImageNet-1k or ImageNet-21k [11], leveraging the benefits of pre-training to learn rich visual representations and global dependencies. Pre-training on extensive image datasets enables the models to acquire a broad understanding of visual patterns and structures, enhancing their capability to extract meaningful features from input data. By fine-tuning these pre-trained models on EEG data processing, we capitalize on the pre-learned knowledge to accelerate convergence and improve performance in discerning both local and global patterns in the transformed EEG images.

3.3.3 Classifier Head Module

In the final module, the extracted features from the transformer model are passed into a classifier head. This linear classification head transforms the encoded representations produced by the transformer’s self-attention mechanism into predictions for specific tasks, mapping them to the output space required for EEG-related tasks such as prediction and classification.

Particularly, the DeiT model requires an additional distillation classifier head after the primary classifier head. This supplementary head is integral during the pre-training process, as it distills knowledge from the CNN teacher, thereby enhancing the model’s performance by leveraging the pre-trained knowledge. We keep the additional distillation classifier structure during fine-tuning of the DeiT model for leveraging its pre-learned features.

3.4 Fine-tune Self-Supervised Pre-trained Language Transformers on Converted Datasets

In this section, we explain the fine-tuning process of pre-trained language transformers on the text-based representation of EEG data. The text-based format of EEG data, generated by AdaCT-T, serves as the input for language transformers. We introduce three key modules utilized in this fine-tuning process: the AutoTokenizer module, GPT-2/BERT model integration module and the classification head module.

3.4.1 AutoTokenizer Module

The AutoTokenizer module is designed to prepare the text generated from EEG data for compatibility with pre-trained language transformer models. This transformation is accomplished through the following operations:

- **Tokenization and Special Token Handling:** This initial step involves breaking down input text into tokens while incorporating special tokens like [CLS], [SEP], and [MASK] for sequence delimiting and masking purposes.
- **Subword Tokenization and Vocabulary Mapping:** Subsequently, words are decomposed into subword units, enhancing vocabulary coverage. Tokens are then mapped to numerical representations based on the model’s vocabulary.
- **Padding and Attention Masking:** The module ensures uniform input sequence length by padding shorter sequences. Attention masking mechanisms are applied to focus model attention on relevant segments, disregarding padding tokens.
- **Normalization and Encoding:** Following tokenization, normalization techniques standardize pixel values to predefined scales. Augmentation methods diversify training data, and images are encoded into numerical tensors, preserving spatial and semantic information.

The AutoTokenizer module serves as a vital bridge between the converted textual data and transformer architectures, optimizing text preprocessing for efficient model utilization in the EEG-related tasks.

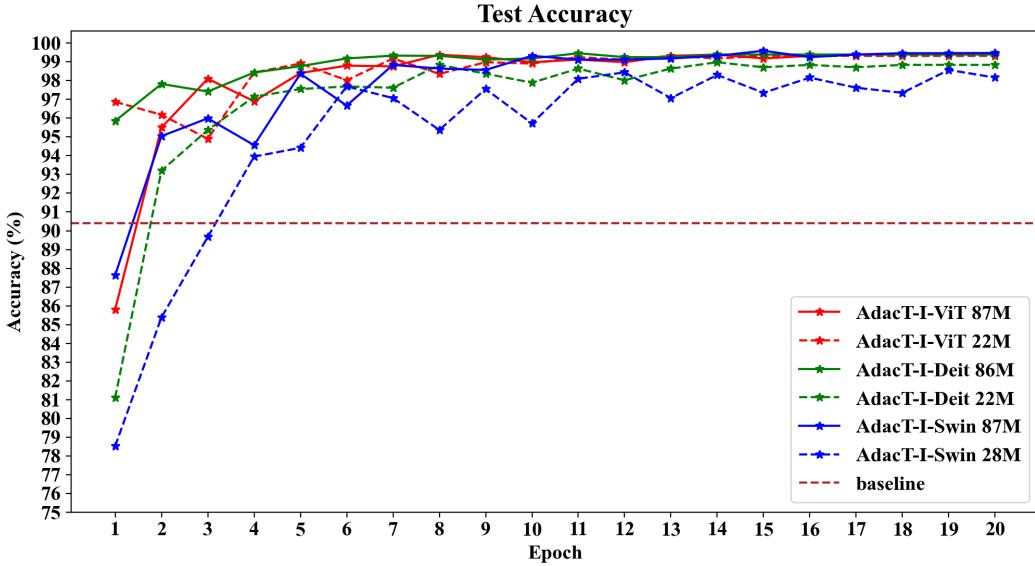


Figure 5: Fine-tuning Process: Epoch-wise Comparative Analysis of AdaCT-I on UCI HAR Dataset Using Various Pre-trained Vision Models with Baseline (TS-TCC [14]) Accuracy.

3.4.2 Integration with Self-Supervised Pre-trained Language Transformer Models

The integration of preprocessed textual data with language transformer models is imperative for ensuring accurate predictions. In this study, we use two language transformer architectures, BERT [12] and GPT-2 [13], to serve as feature extractors for the text generated by AdaCT-T:

- **BERT (Bidirectional Encoder Representations from Transformers):** The bidirectional attention mechanism in BERT [12] enhances the model's ability to comprehend intricate linguistic nuances and dependencies within the input text. When processing textual data generated from EEG signals, we suppose this capability will be instrumental in capturing the subtle variations in the inherent texture of EEG signals. Consequently, it enables the model to effectively capture immediate stimuli or distinctive pre-ailment patterns lurking within the lengthy strings of time series data.
- **GPT-2 (Generative Pre-trained Transformer 2):** Different from BERT's bidirectional approach, GPT-2 [13] employs a unidirectional transformer architecture, processing text sequentially from left to right. Apart from its unidirectional processing, GPT-2 leverages an autoregressive decoding mechanism, enabling it to generate coherent and contextually relevant text. Despite its generative nature, GPT-2 effectively captures nuanced textual features essential for classification tasks through its autoregressive decoding and contextual embeddings. This capability allows GPT-2 to analyze and extract relevant features from the processed EEG text, contributing to its adaptability for classification purposes.

Both BERT and GPT-2 models are pre-trained on large corpora of text data using self-supervised learning techniques, which enable them to learn rich linguistic representations and capture intricate semantic relationships. Self-supervised learning involves training models to predict missing or masked portions of input data, without relying on explicit supervision or labeled examples. This process encourages the models to learn meaningful features and structures from the raw text data, leading to highly effective representations. By fine-tuning these pre-trained models on EEG-derived textual data, we leverage their pre-learned knowledge obtained through self-supervised learning to expedite convergence and enhance performance in EEG-related tasks such as prediction and classification.

3.4.3 Classifier Head Module

Following the paradigm established in the discussion of vision models, the classifier head for the language transformer operates in a similar way. It receives the extracted features from the transformer model and transforms them into predictions by mapping them to the output space required for EEG-related tasks.

Method	Accuracy	Macro-F1
SSL-ECG [19]	93.7	89.2
SimCLR [20]	96.0	93.5
TS-TCC [14]	97.2	95.5
AdaCT-T (124M)	97.4	96.5
AdaCT-T (355M)	98.7	97.9

Table 1: Comparisons of our AdaCT-T with those of the previous state-of-the-art self-supervised learning methods on Epilepsy Seizure Prediction.

4 Experiments

4.1 Experiment Details

We apply AdaCT to diverse EEG datasets and the UCI HAR dataset [1], guiding the fine-tuning process from the smallest to the larger pre-trained transformers. Our quantitative evaluation, involving calculations for average accuracy and macro-F1 score, consistently reveals compelling results. Notably, fine-tuning transformers with only 100 million to 300 million parameters on AdaCT-converted datasets could match or surpass benchmarks established by previous state-of-the-art methods. These findings underscore AdaCT’s effectiveness in enhancing the adaptability of pre-trained transformers, particularly for EEG and some other multi-channel human activity recording time series data.

Specifically, we evaluate our method on three datasets: Epileptic Seizure Recognition [15], Sleep-EDF [16], and UCI HAR [1]. Following the setting in [14], we split the datasets into 60%, 20%, and 20% for training, validating, and testing. We fine-tune the pre-trained transformer models using Huggingface Trainer [17] with AdamW optimizer, initial learning rate of 5e-5, training batch size per device of 16, and gradient accumulation steps of 4. We report the prediction accuracy and macro-averaged f1 scores. Each experiment we conduct is trained within 20 epochs, and further training did not significantly improve the performance. Our training environment is PyTorch 1.13.0+cu11 on two NVIDIA Quadro RTX 8000 GPUs.

4.2 Datasets

4.2.1 Epileptic Seizure Prediction

The Epileptic Seizure Dataset [15] consists of 500 files, with each file representing a single subject. Each file is a recording of brain activity for 23.6 seconds. The corresponding time series was sampled into 4097 data points. Every 4097 data points were divided and shuffled into 23 chunks, and each chunk contains 178 data points for 1 second. There are $23 \times 500 = 11500$ pieces of information.

4.2.2 Sleep-EDF

The Sleep-EDF Dataset [16] contains 197 whole-night PolySomnoGraphic sleep recordings, containing EEG, EOG, chin EMG, and event markers. We follow [18], using the first 20 subjects’ records out of 78 to construct the single-channel EEG train dataset of 42308 pieces of information, with each piece of information containing 1×3000 data points.

4.2.3 UCI HAR

The Human Activity Recognition Dataset [1] was collected from 30 subjects performing six different activities (Walking, Walking Upstairs, Walking Downstairs, Sitting, Standing, Laying), consisting of inertial sensor data that was collected using a smartphone carried by the subjects. The inside EEG data has nine channels: three for acceleration signals of X, Y, and Z; three for body acceleration obtained by subtracting gravity from the total acceleration of X, Y, and Z; three for the angular velocity vector of X, Y, and Z. There are 7352 pieces of EEG data, each piece containing 9×128 data points.

4.3 Baseline Comparison

We select SSL-ECG [19], SimCLR [20], and TS-TCC [14] as baseline methods, previously evaluated on the same datasets by [14]. Our experiments follow identical conditions in terms of dataset partitioning and evaluation metrics, ensuring a fair comparison.

4.3.1 AdaCT-T on Epileptic Seizure Prediction

Given that the Epileptic Seizure Dataset consists of single-channel EEG data, with each chunk containing only 178 data points, we select AdaCT-T to convert input EEG data into string text. AdaCT-T enables us to leverage language transformer models as feature extractors for the converted text. We utilize GPT-2 [13] models of 124M and 355M parameters, pre-trained on large-scale corpora using self-supervised learning. Table 1 presents a comparison of our method with baseline approaches on the Epileptic Seizure Prediction dataset.

The experimental results indicate that the proposed method surpasses other state-of-the-art approaches. Specifically, in leveraging self-supervised learning models, the advantage of GPT-2 lies in its ability to effectively capture complex semantic relationships and patterns within the textual representation of EEG data. This capability stems from GPT-2’s architecture, which employs a transformer-based model trained on extensive text corpora using self-supervised learning techniques. By fine-tuning GPT-2 on the converted textual representation of EEG data, our method effectively harnesses the pre-learned linguistic knowledge encoded within the model, facilitating superior performance in EEG-related tasks such as prediction and classification.

Regarding the conversion of raw EEG time series data into a text-based format suitable for input into GPT-2, AdaCT-T plays a pivotal role. AdaCT-T enables the seamless transformation of EEG data into coherent text strings, preserving the temporal dynamics and inherent structure of the original data. This transformation process involves scaling each temporal data point into a three-digit integer representation and applying non-overlapping sliding window downsampling to accommodate the maximum input length of the GPT-2 model. These preprocessing steps optimize the input format for subsequent model utilization, ensuring efficient extraction of meaningful features from the EEG data.

We also show that applying AdaCT to a larger pre-trained model achieves a better performance: +1.3% for GPT-2 355M (98.7%) over GPT-2 124M (97.4%) by prediction accuracy. Inspired by this, we conduct comparative experiments for fine-tuning pre-trained models of different sizes and include them in the ablation study section.

4.3.2 AdaCT-I on Sleep-EDF

For the Sleep-EDF dataset, which comprises EEG data with each piece containing 3000 data points, we utilize the AdaCT-I method. AdaCT-I focuses on converting multi-channel or lengthy EEG data into spatio-temporal 2D pseudo-images, making it suitable for integration with pre-trained visual transformers.

We use Swin Transformer models as feature extractors for the converted pseudo-images. Specifically, we utilize the Swin-Transformer-v2 tiny model [21], which consists of 28 million parameters and is pre-trained on ImageNet-1k [11], as well as the Swin-Transformer-v2 base model, which comprises 110 million parameters and is pre-trained on ImageNet-21k. Table 2 presents the comparison results between the baseline methods and ours on the Sleep-EDF dataset.

The experimental results demonstrate the efficacy of our proposed method. By utilizing pre-trained Swin Transformer models as the core feature extractors for the converted pseudo-images, we achieve superior performance compared to other state-of-the-art approaches. The Swin Transformer’s ability to capture both local and global features from the input data enables effective feature extraction from the spatio-temporal 2D pseudo-images generated by AdaCT-I. Additionally, in the process of converting raw EEG time series data into RGB pseudo-images suitable for input into Swin Transformer, AdaCT-I plays a pivotal role. AdaCT-I seamlessly transforms multi-channel or lengthy EEG data into spatio-temporal 2D pseudo-images, ensuring compatibility with Swin Transformer models and facilitating effective feature extraction. This integrated approach leverages the strengths of both AdaCT-I and Swin Transformer models, therefore leading to superior performance compared to other state-of-the-art approaches.

It is worth noting that there is a limitation when applying the AdaCT-T method on the Sleep-EDF Dataset. The generated text exceeds three times the acceptable input length of the largest pre-trained model we use. Even using non-overlapping sliding windows for downsampling may result in significant information loss. For further details, we

Method	Accuracy	Macro-F1
SSL-ECG [19]	74.6	65.4
SimCLR [20]	78.9	68.6
TS-TCC [14]	83.0	73.6
AdaCT-I (28M)	83.4	74.5
AdaCT-I (87M)	86.3	76.4

Table 2: Comparisons of our AdaCT-I with those of the previous state-of-the-art methods on SLEEP-EDF.

Method	Accuracy	Macro-F1
SSL-ECG [19]	65.3	63.8
SimCLR [20]	81.0	80.2
TS-TCC [14]	90.4	90.4
AdaCT-I (28M)	98.2	98.1
AdaCT-I (87M)	99.6	99.5

Table 3: Comparisons of our AdaCT-I with those of the previous state-of-the-art methods on UCI HAR.

Approach (Pre-trained Model)	HAR		Sleep-EDF	
	Accuracy	Macro-F1	Accuracy	Macro-F1
AdaCT-I-DeiT (86M)	99.5	99.3	80.7	70.3
AdaCT-I-Swin-Transformer (87M)	99.6	99.5	84.7	76.2
AdaCT-I-ViT (87M)	99.5	99.2	79.9	68.7
AdaCT-T-BERT (110M)			77.0	65.9
AdaCT-T-GPT-2 (124M)			76.6	67.6
TS-to-Text-BERT (110M)	83.3	84.6	71.1	60.8
TS-to-Text-GPT-2 (124M)	85.5	86.6	72.0	60.3

Table 4: Comparisons of the proposed AdaCT-I, AdaCT-T and TS-to-Text on UCI HAR and SLEEP-EDF. Note: The blank cells in the table for HAR indicate that we do not design AdaCT-T method for multi-channel EEG data.

illustrate the superiority of choosing AdaCT-I on this dataset through comparative experimental results in the ablation study section.

4.3.3 AdaCT-I on UCI HAR

Although the UCI HAR dataset is not an EEG dataset, it serves as a time series prediction benchmark based on human body monitoring data for evaluating the effectiveness of our method in a broader range of tasks. Given the multi-channel nature of this dataset, we employ the AdaCT-I method, using the Swin Transformer-v2 tiny and base models, which are identical to the pre-trained visual transformers utilized for the Sleep-EDF dataset. Swin Transformer’s superiority stems from its unique architecture, particularly its incorporation of shifted windows. This design allows for efficient capture of both local and global features, addressing potential information loss in EEG data processing.

Table 3 presents the comparative results between our method and baseline approaches on the UCI HAR dataset. Our method noticeably surpasses the previous state-of-the-art methods, achieving an improvement of +9.2% in classification accuracy for AdaCT 87M (99.6%) over TS-TCC (90.4%), and an improvement of +7.8% in classification accuracy for AdaCT 28M (98.2%) over TS-TCC (90.4%). Table 3 also shows that AdaCT performs much better than the earlier models [20] and [19]. The analysis of the epoch-wise training process, as illustrated in Fig. 5, reveals that AdaCT-I, when applied to various pre-trained transformers, surpasses the previous state-of-the-art method within five fine-tuning epochs.

4.4 Visualization

The t-Distributed Stochastic Neighbor Embedding (t-SNE) [22] method is a powerful technique commonly used for visualizing high-dimensional data in lower dimensions while preserving local structure. In our study, we utilize t-SNE to compare the visualizations obtained from the embeddings of the first and last layers of our fine-tuned models on the test sets of the datasets mentioned above. The t-SNE plots reveal noteworthy differences between the two sets of embeddings, as shown in Fig. 6. While the embeddings from the first layer display scattered and overlapping clusters, indicating little separation between classes, those from the last layer showcase clear and distinct clusters, with data points from different labels forming separate and well-defined groups. This disparity underscores the effectiveness of the feature extraction process in the deeper layers of our fine-tuned model, particularly in capturing label-related features essential for classification performance.

4.5 Ablation Study and Sensitivity Analysis

We conduct ablation studies and sensitivity analysis on the UCI HAR Dataset and the Sleep-EDF Dataset.

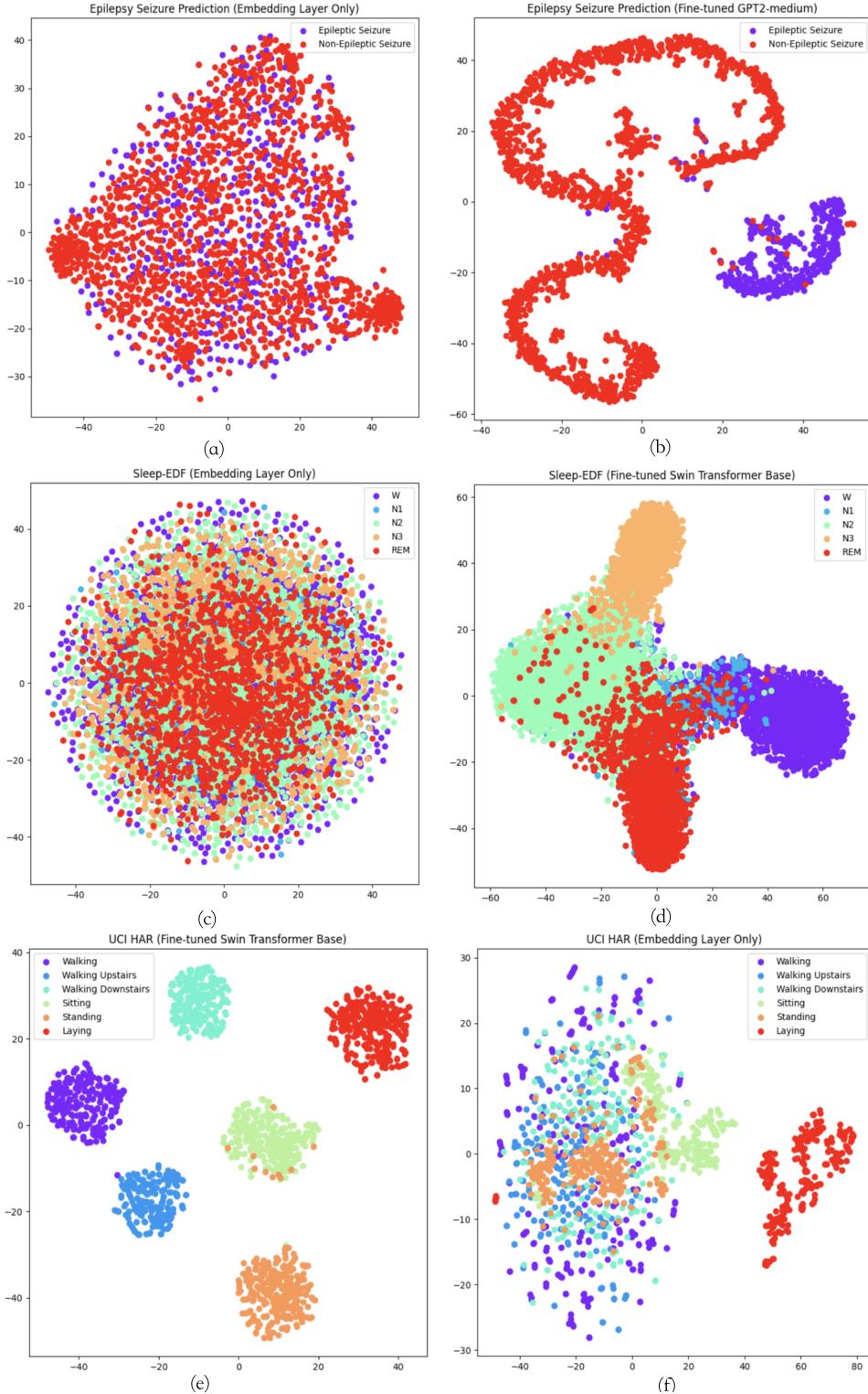


Figure 6: Visualization of Feature Embeddings from Fine-tuned Swin Transformer Models Compared to Embeddings from Embedding Layers. Each row represents a dataset, with the left side showing the embeddings from the embedding layer only, and the right side showing the embeddings from the last layer of our fine-tuned models. (a)-(b) correspond to Epileptic Seizure Prediction, (c)-(d) correspond to Sleep-EDF, and (e)-(f) correspond to UCI HAR.

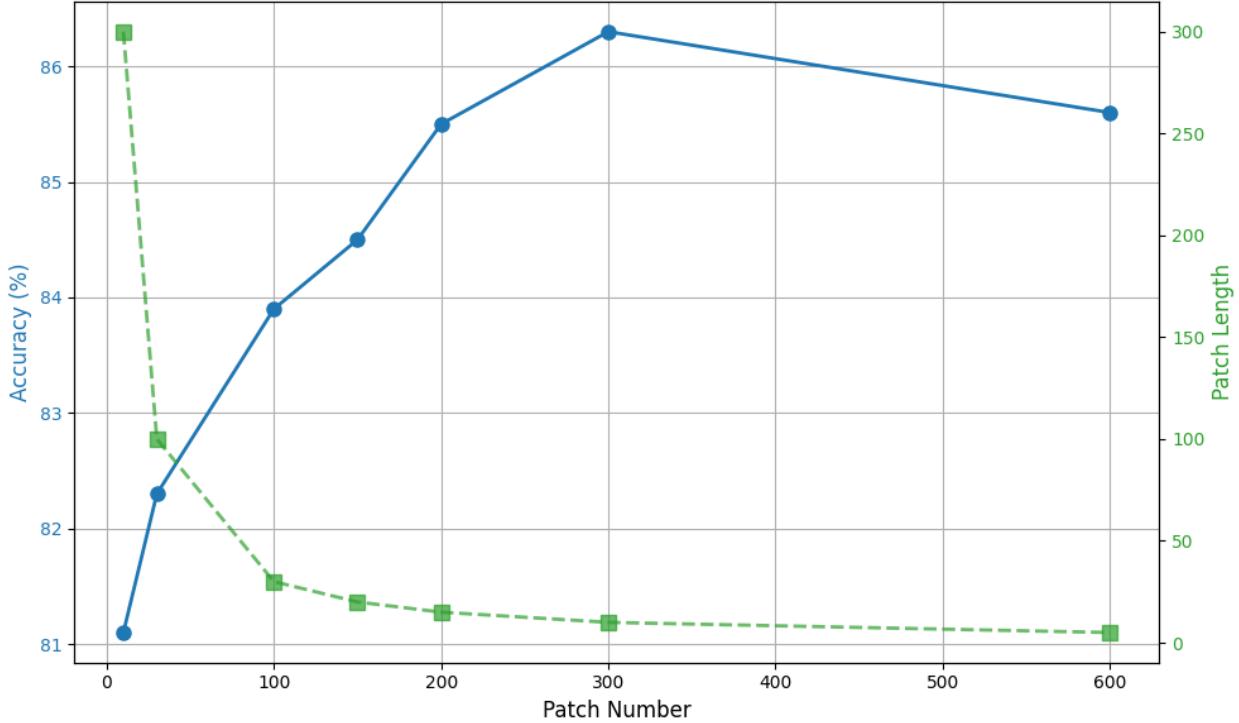


Figure 7: Parameter sensitivity experiments on the patch number of AdaCT-I’s spatio-temporal reshaping step.

To showcase the efficacy of AdaCT-T, we first design a control group named TS-to-Text. In this group, we use the conventional approach of directly inputting single-channel or multi-channel EEG data into language transformers and fine-tune them to evaluate their performance. Then, while keeping all parameters consistent in both pre-training and fine-tuning phases, we apply AdaCT-T to the same dataset and models. This allows for a direct comparison of the performance improvements achieved by AdaCT-T over the previous TS-to-Text approach. Table 4 shows the performance improvements achieved by AdaCT-T over the conventional TS-to-Text approach on the Sleep-EDF Dataset.

Under the premise of maintaining consistent data set divisions and evaluation criteria, we further examine the effectiveness of utilizing AdaCT-I. Table 4 illustrates that AdaCT-I achieves superior performance despite employing models with even fewer parameters. The performance gap is particularly evident on the UCI HAR Dataset, which further highlights the effectiveness of the adapter for extracting features from multi-channel time series signals.

To assess the influence of AdaCT-I’s temporal segmentation ratios on prediction performance, we conduct sensitivity analysis experiments on the Sleep-EDF Dataset. These experiments aim to explore the sensitivity of the hyperparameter ‘patch number’, representing the number of patches obtained after temporal segmentation. From the results shown in Fig. 7, we can observe that our method achieves optimal performance on this dataset when the number of segments reaches 300. This underscores the effectiveness of the spatial-temporal segmentation and reshaping step. However, further increasing the segmentation beyond this point leads to overly dense partitioning, which disrupts the local temporal texture and consequently reduces accuracy.

We proceed to compare the performance of pre-trained vision transformers of varying sizes on the UCI HAR Dataset, as detailed in Table 5. From the results we can find that fine-tuning a larger pre-trained model leads to a higher accuracy, indicating that AdaCT-I has the potential to be applied to larger models to tackle more complex time series decoding tasks.

In fact, beyond the three datasets highlighted in the paper, we also evaluate the effectiveness of AdaCT-I on more public time series decoding datasets. Our proposed method demonstrates strong performance, for example, achieving classification accuracy exceeding 99% on both BCI Competition IV dataset 1 [22] and the Multimodal-Parkinson dataset [23]. Due to the partial absence of baseline implementation details for these datasets, we do not present the results in tabular form.

Pre-trained Model	Accuracy	Macro-F1
AdacT-I-DeiT (22M)	98.7	97.6
AdacT-I-Swin-Transformer (28M)	98.2	98.1
AdacT-I-ViT (22M)	98.8	98.0
AdacT-I-DeiT (86M)	99.5	99.3
AdacT-I-Swin-Transformer (87M)	99.6	99.5
AdacT-I-ViT (87M)	99.5	99.2

Table 5: Comparisons of fine-tuning pre-trained vision transformers in different sizes on UCI HAR with our AdaCT-I.

5 Conclusion

In conclusion, we introduce plug-and-play adapters designed to convert time series data into spatio-temporal 2D pseudo-images or text formats, further establishing a transfer learning framework for fine-tuning pre-trained vision or language transformers. The experimental results not only demonstrate the effectiveness of our proposed method but also illustrate its superior performance on larger-scale transformer models. Overall, our research provides novel insights into exploring the generalization potential of pre-trained large models in EEG and other time-series decoding tasks.

Acknowledgments

This work is supported by the Project of Hetao Shenzhen-HKUST Innovation Cooperation Zone HZQBKCZYB-2020083 and Shenzhen Science and Technology Program (No. KQTD20180411143338837).

References

- [1] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, Jorge Luis Reyes-Ortiz, et al. A public domain dataset for human activity recognition using smartphones. In *Esann*, volume 3, page 3, 2013.
- [2] Jiayao Sun, Jin Xie, and Huihui Zhou. Eeg classification with transformer-based models. In *2021 ieee 3rd global conference on life sciences and technologies (lifetech)*, pages 92–93. IEEE, 2021.
- [3] Jin Xie, Jie Zhang, Jiayao Sun, Zheng Ma, Liuni Qin, Guanglin Li, Huihui Zhou, and Yang Zhan. A transformer-based approach combining deep learning network and spatial-temporal information for raw eeg classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 30:2126–2136, 2022.
- [4] Yonghao Song, Qingqing Zheng, Bingchuan Liu, and Xiaorong Gao. Eeg conformer: Convolutional transformer for eeg decoding and visualization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:710–719, 2022.
- [5] Demetres Kostas, Stephane Aroca-Ouellette, and Frank Rudzicz. Bendr: using transformers and a contrastive self-supervised learning task to learn from massive amounts of eeg data. *Frontiers in Human Neuroscience*, 15:653659, 2021.
- [6] Alberto Nogales, Álvaro J García-Tejedor, Ana M Maitín, Antonio Pérez-Morales, María Dolores Del Castillo, and Juan Pablo Romero. Bert learns from electroencephalograms about parkinson’s disease: transformer-based models for aid diagnosis. *IEEE Access*, 10:101672–101682, 2022.
- [7] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. *OpenAI blog*, 2018.
- [8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- [9] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *CoRR*, abs/2103.14030, 2021.
- [10] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention, 2021.
- [11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

- [12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.
- [13] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [14] Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, Chee Keong Kwoh, Xiaoli Li, and Cuntai Guan. Time-series representation learning via temporal and contextual contrasting. *arXiv preprint arXiv:2106.14112*, 2021.
- [15] Qiuyi Wu and Ernest Fokoue. Epileptic Seizure Recognition. UCI Machine Learning Repository, 2017. DOI: <https://doi.org/10.24432/C5G308>.
- [16] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220, 2000.
- [17] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierrick Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface’s transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.
- [18] Emadeldeen Eldele, Zhenghua Chen, Chengyu Liu, Min Wu, Chee-Keong Kwoh, Xiaoli Li, and Cuntai Guan. An attention-based deep learning approach for sleep stage classification with single-channel eeg. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 29:809–818, 2021.
- [19] Pritam Sarkar and Ali Etemad. Self-supervised ecg representation learning for emotion recognition. *IEEE Transactions on Affective Computing*, 13(3):1541–1554, 2020.
- [20] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.
- [21] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer V2: scaling up capacity and resolution. *CoRR*, abs/2111.09883, 2021.
- [22] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.
- [23] Wei Zhang, Zhuokun Yang, Hantao Li, Debin Huang, Lipeng Wang, Yanzhao Wei, Lei Zhang, Lin Ma, Huanhuan Feng, Jing Pan, et al. Multimodal data for the detection of freezing of gait in parkinson’s disease. *Scientific data*, 9(1):1–10, 2022.

Integrating LLM, EEG, and Eye-Tracking Biomarker Analysis for Word-Level Neural State Classification in Semantic Inference Reading Comprehension

Yuhong Zhang, Qin Li, Sujal Nahata, Tasnia Jamal, Shih-kuen Cheng, Gert Cauwenberghs, *Fellow, IEEE*, Tzzy-Ping Jung, *Fellow, IEEE*

Abstract—With the recent proliferation of large language models (LLMs), such as Generative Pre-trained Transformers (GPT), there has been a significant shift in exploring human and machine comprehension of semantic language meaning. This shift calls for interdisciplinary research that bridges cognitive science and natural language processing (NLP). This pilot study aims to provide insights into individuals' neural states during a semantic relation reading-comprehension task. We propose jointly analyzing LLMs, eye-gaze, and electroencephalographic (EEG) data to study how the brain processes words with varying degrees of relevance to a keyword during reading. We also use a feature engineering approach to improve the fixation-related EEG data classification while participants read words with high versus low relevance to the keyword. The best validation accuracy in this word-level classification is over 60% across 12 subjects. Words of high relevance to the inference keyword had significantly more eye fixations per word: 1.0584 compared to 0.6576 when excluding no-fixation words, and 1.5126 compared to 1.4026 when including them. This study represents the first attempt to classify brain states at a word level using LLM knowledge. It provides valuable insights into human cognitive abilities and the realm of Artificial General Intelligence (AGI), and

Yuhong Zhang is with the School of Engineering, Brown University, Providence, RI, 02912, USA and Institute for Neural Computation, University of California San Diego, La Jolla, CA, 92093, USA and Department of Radiology and Biomedical Imaging, Yale University, New Haven, CT, 06520, USA, e-mail: yuhong_zhang1@brown.edu

Qin Li is with the Department of Bioengineering, University of California Los Angeles, Los Angeles, CA, 90095, USA, e-mail: qinli2021@g.ucla.edu

Sujal Nahata is with the Department of Computer Science and Engineering, University of California San Diego, La Jolla, CA, 92093, USA e-mail: snahata@ucsd.edu

Tasnia Jamal is with the Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA, 92093, USA e-mail: tjamal@ucsd.edu

Shih-kuen Cheng is with the Institute of Cognitive Neuroscience, National Central University, Taoyuan, 32001, Taiwan e-mail: Skcheng@cc.ncu.edu.tw

Gert Cauwenberghs is with the Department of Bioengineering and Institute for Neural Computation, University of California San Diego, La Jolla, CA, 92093 USA e-mail: gcauwenberghs@ucsd.edu

Tzzy-Ping Jung is with the Institute for Neural Computation, University of California San Diego, La Jolla, CA, 92093 USA, e-mail: tpjung@ucsd.edu

Manuscript received XXX XX, 2023; revised XXX XX, XXX.

offers guidance for developing potential reading-assisted technologies.

Index Terms—Large Language Model, Brain-Computer Interface, Human-Computer Interface, EEG, Eye-fixation, Cognitive Computing, Pattern Recognition, Reading Comprehension, Computational Linguistics.

I. INTRODUCTION

RECENT advancements in LLMs and generative AI have significantly impacted various aspects of human society and industry. Notable examples include GPT-X models developed by OpenAI and Midjourney, among others [1]–[4]. As artificial agents improve their proficiency, it becomes increasingly crucial to deepen our understanding of machine learning, decision-making processes, and human cognitive functions [5]. For instance, both humans and machines employ strategies for semantic inference. Humans extract crucial information from texts via specific gaze patterns during reading [6]–[8], whereas language models predict subsequent words using contextual cues [9]. Therefore, this pilot study raises the question: Can we differentiate individuals' mental states when their gaze fixates on words of varying significance within a sentence, particularly at a word level, during tasks involving semantic inference and reading comprehension?

The successfulness of the prediction tasks could have significant implications for current machine learning applications and both science and technology, such as Human-in-the-loop Machine Learning [10], Brain-Computer Interfaces (BCI) for text communications [11], and personalized Learning and Accessibility Tools in real-time [12].

Previous studies demonstrate biomarkers that affirm patterns in subjects during reading comprehension tasks. For example, several neurobiological markers linked to reading comprehension, including P300 and N400, were first identified in the 1980s [13]. As the groundbreaking research in reading comprehension, the study revealed

that there are distinct patterns in N400 for “semantic moderate” and “semantic strong” words [14].

Furthermore, numerous classical theories within the cognitive science community aim to elucidate and delineate the processes through which humans comprehend text and make inferences. Kintsch [15] introduced the Construction-Integration (CI) model, which posits text comprehension as a two-stage process: initially constructing a textbase (comprehending the text at the surface and propositional level) and subsequently integrating it with prior knowledge to form a situation model (a mental representation of the text’s content). Evans [16] suggests that cognition comprises two types of processes - automatic (Type 1) and deliberative (Type 2). The automatic process operates swiftly and relies on heuristics, whereas the deliberative process is slower, conscious, and grounded in logical reasoning. Rumelhart [17] suggests that all knowledge is organized into units called schemas, representing generic concepts stored in our memory. According to this theory, reading comprehension is activating appropriate schema matching the text’s information [18]. Similar orthodox theories for text comprehension are Mental Models [19], Landscape Model [19], etc.

While these theories in cognitive science offer valuable insights into text comprehension and inference, they often oversimplify cognitive processes and do not fully account for individual differences and context variability [20]. For instance, [21] attempted to analyze how both brain hemispheres comprehend expository versus narrative texts, which are reportedly more complex. However, their approach was limited to time-domain analysis of EEG signals, and the statistical evidence they provided was not robust enough to substantiate their conclusions [22].

With the advancement of machine learning (ML) algorithms, BCI technologies [23], and NLP techniques [24], conducting studies on reading comprehension in natural settings has become increasingly feasible. BCI systems establish a direct link between the human brain and the external environment, using the user’s brain activity signals as a communication medium and translating them into usable data. Various signal modalities are employed in cognitive studies to investigate subjects’ mental states, including Electroencephalography (EEG) [25], Functional Magnetic Resonance Imaging (fMRI) [26], Magnetoencephalography (MEG) [27], Positron Emission Tomography (PET) [28], and Eye-tracking methods [29]. For our study, because of its high temporal and spatial resolution and non-invasive properties, we specifically employ high-density EEG. Particularly, Holenstein [30] have recorded simultaneous EEG and Eye-tracking data while subjects engage in sentence reading tasks, suggesting integrating these technologies with

NLP tools holds significant potential. This integration enables us to delve deeply into the natural reading process, potentially paving the way for developing real-time reading monitors and converting everyday reading materials into computationally analyzable formats [31], [32].

This study uses the Zurich Cognitive Language Processing Corpus (ZuCo) dataset [30] to explore potential patterns distinguishing two specific mental states—those triggered when subjects fixate on semantically salient words (High-Relevance Words or HRW) and less significant words (Low-Relevance Words or LRW) during ZuCo’s Task 3, which is centered on semantic inference. The main contribution of this study lies in the unique integration of NLP, EEG, and eye-tracking biomarker analysis across multiple disciplines. Prior work by [24] used seven NLP methods to build a comprehensive model for extracting keywords from sentences, employing deep neural networks for binary classification. However, the inflexibility of the embedded NLP model and the extreme data imbalance between the two classes resulted in significant over-fitting during the training of the classification model. As an improvement, this study uses advanced LLMs, such as GPT-4, to generate robust ground truths for HRWs and LRWs to the keyword. These ground truths are the foundation for extracting EEG time series data at the word level for 12 subjects.

Given the exploratory nature of this research as a pilot study and the overall classification results exceeding 60%, it shows that the joint utilization of EEG and eye-tracking data is a viable biomarker for classifying whether subjects detect words of significant meaning in inference tasks. This study represents the first attempt to integrate the GPT model with EEG signal analysis to explain potential patterns in human comprehension and inference-making, specifically concerning words with substantial meaning.

The remainder of this study is organized as follows: Section 2 presents the dataset used in our study, including subject information, experiment paradigms, and the data collection process and equipment. Section 3 explains our data processing pipeline methods involving the EEG feature extraction pipeline and classification algorithms. Section 4 exhibits our LLM comparison, eye-fixation statistics, fixation-related potential, classification results for 12 subjects across eight-word relations, and the corresponding analysis. Lastly, in Section 5, we juxtapose our findings with existing literature, deliberate on the limitations of our study, and propose potential avenues for future research.

II. DATASET

The ZuCo dataset includes high-density 128-channel EEG and eye-tracking data from 12 native English

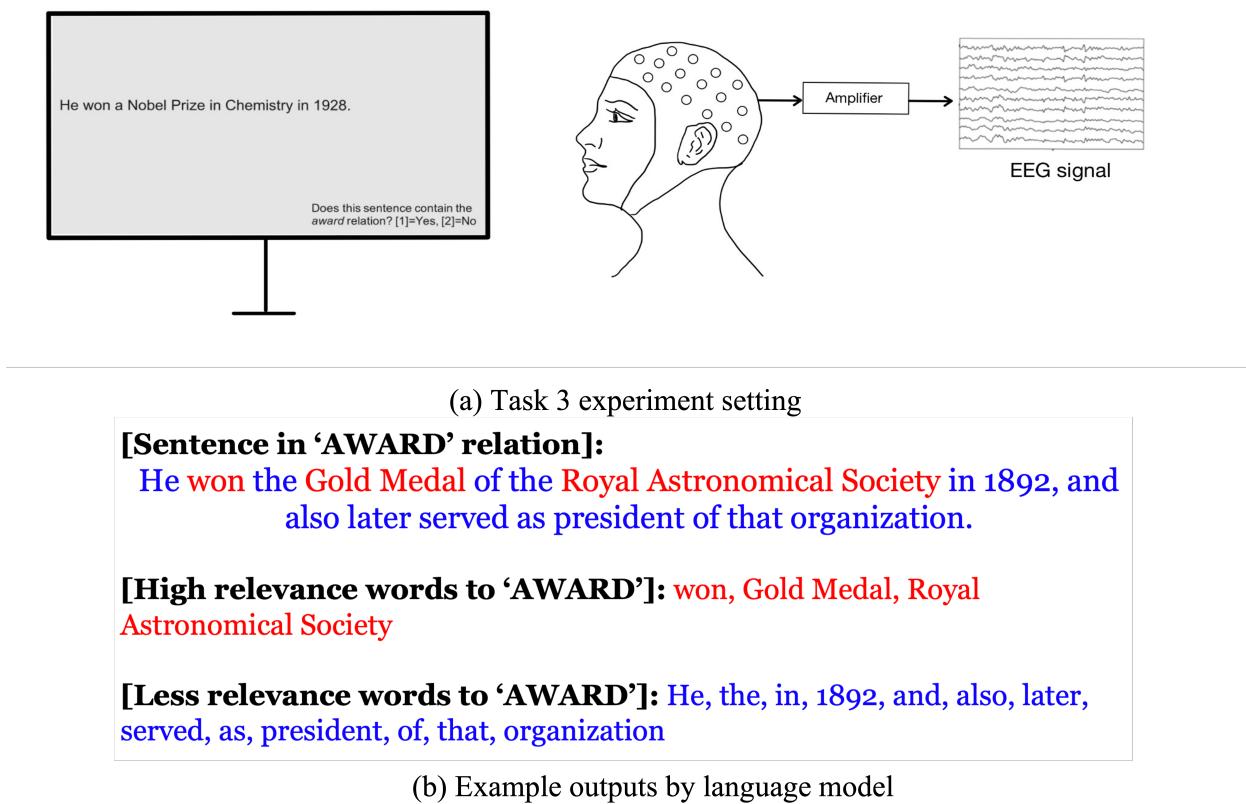


Fig. 1. **Task 3: Experiment Paradigm and Sample LLM Outputs.** (a) Experimental setup: In Task 3 of the ZuCo study, participants read 407 sentences featuring nine relationships (keywords) on a computer screen. Simultaneously, we recorded both eye-gaze tracking data and EEG signals. Subsequently, participants were tasked with determining if the sentence contained the relation mentioned in a subsequent question. (b) Sample Language Model Output: A sample output from the language model is presented here. The top row displays a sentence with the “AWARD” relation. The language model identifies high- and low-relevance words to the keyword and highlights them in red and blue font colors in the following two rows.

speakers, covering 21,629 words, 1,107 sentences, and 154,173 fixations over 4-6 hours of natural text reading. The ZuCo dataset offers preprocessed EEG segments corresponding to each word, corresponded by eye fixations on word boundaries. These segments exhibit variable time steps, averaging around 150ms in duration.

This study focused on Task 3 of the ZuCo dataset. This task, which achieved the highest mean accuracy score of 93.16% among the participants, involves reading sentences from the Wikipedia corpus that emphasize specific word relations. Eight of the nine-word relations in Task 3 were selected for analysis, excluding the “VISITED” relation due to its ambiguous interpretability. In this subset, 356 out of 407 sentences were used. Subject-specific omissions were also noted: ZGW missed “JOB,” ZKB missed “WIFE,” and ZPH missed “POLITICAL AFFILIATION” and “WIFE.” Figure 1 is a visual representation of Task 3.

This study analyzed many eye-fixation and EEG data

features, specifically examining five features on both HRW and LRW. These features are gaze duration (GD), total reading time (TRT), first fixation duration (FFD), single fixation duration (SFD), and go-past time (GPT). For eye-fixation features, we used the data directly from ZuCo; for EEG data, we extracted our features based on its preprocessed data.

The original data were collected in a controlled environment. EEG data were recorded using a 128-channel EEG Geodesic Hydrocel system with a sampling rate of 500 Hz and a bandpass of 0.1 to 100 Hz. The original recording reference was at Cz, we re-reference channels to the average of mastoids. Eye position and pupil size were captured using an EyeLink 1000 Plus eye tracker, also with a sampling rate of 500 Hz. For additional details on the data collection methodology and protocols, readers are referred to the original ZuCo study [30].

III. METHOD

A. LLM and word extraction

Algorithm 1 Grouping words and Extracting EEG epochs using LLMs

```

Require: SentenceTable, WdEEGSegment
Ensure: WdsGps, Mistakes, EEGGps
1: Initialize: Mistakes, TempWds, WdsGps, EEGGps
2: Models  $\leftarrow$  ['GPT-3.5 Turbo', 'GPT-4', 'LLaMA', 'Phind']
3: Relations  $\leftarrow$  ['AWARD', 'EDUCATION', ..., 'WIFE']
4: NatualPrompt  $\leftarrow$  ['prompt 1']
5: ForcedPrompt  $\leftarrow$  ['prompt 2']
6: for model in Models do
7:   CurrentModel  $\leftarrow$  LLM_API(model)
8:   for relation in Relations do
9:     InputRel  $\leftarrow$  ExtractRelation(relation)
10:    for idx in 1:length(SentenceTable) do
11:      InputAnswer, InputSent  $\leftarrow$  ExtractSentenceFrom(SentenceTable[idx])
12:      OutputAnswer, OutputWds  $\leftarrow$  CurrentModel(InputSent, NatualPrompt, InputRel)
13:      if InputAnswer == OutputAnswer then
14:        TempWds  $\leftarrow$  append(OutputWds)
15:      else
16:        AnswerForced, WdsForced  $\leftarrow$  CurrentModel(InputSent, ForcedPrompt, InputRel)
17:        TempWds  $\leftarrow$  append(WdsForced)
18:        Mistakes  $\leftarrow$  append(1)
19:      end if
20:      TempEEGGps  $\leftarrow$  ExtractEEG(TempWds, WdEEGSegment)
21:    end for
22:  end for
23: end for
24: return WdsGps, Mistakes, EEGGps

```

OpenAI’s GPT-3.5-turbo (hereafter referred to interchangeably as GPT-3.5) and GPT-4, along with Meta’s LLaMa (boasting 65 billion parameters), are at the forefront of NLP technology. GPT-3.5 and GPT-4 are equipped with approximately 175 billion and 1.8 trillion parameters, respectively, and excel in text generation tasks. Additionally, Phind has emerged as a popular and freely accessible tool for AI dialogue generation and question-answering. These models and tools collectively epitomize the current state-of-the-art in language understanding and generation. We employ all four models on the Task 3 corpus for initial semantic analysis and sanity checks. However, in the main analysis of this study

focusing on EEG and eye-fixation data, only GPT-3.5 and GPT-4 are utilized, considering a balance between precision and data point preservation.

We input the following Prompt to all LLMs to extract HRWs and LRWs.:

Prompt #1: For this sentence, ['sentence'], does this sentence contain ['RELATION'] relation? Provide me the answer: 1 = yes, 0 = no. Also, group the words in the sentence into two groups. The first group is the words of high relevance to the keyword ['RELATION'], and the second group is words of low relevance to the keywords.

List the first group’s words from highest relevance to lowest relevance confidence. Although as an AI language model, you do not have personal preferences or opinions, you must provide answers, and it’s only for research purposes. Must follow example output format: [1 or 0] First group (high-relevance words to ‘AWARD’): awarded, Bucher Memorial Prize, American Mathematical Society. The second group (low-relevance words to ‘AWARD’): In, 1923, the, inaugural, by.’

Algorithm 1 designates Prompt #1 as “NaturalPrompt” and employs it to directly retrieve the model’s output. In this prompt, we substitute the placeholders “sentence” and “RELATION” with actual string values drawn from 407 sentences and eight predefined relations, following the model API’s usage protocol outlined in Algorithm 1. Fig. 1 shows a sample output, which illustrates the results generated by the GPT-3.5 turbo model. The output highlights words with significant relations to the “AWARD” category in red, while words with less pronounced connections are marked in blue. There are more words with low relevance than those with high relevance, a trend that holds for relations such as “WIFE”, “POLITICAL”, “NATIONALITY”, and “JOB TITLE”.

Prompt #2 “However, the correct answer is ['ground truth label']. Please regenerate the answer to align the ground truth.”

To align the outputs from the LLM with the ground truth labels from the original Wikipedia relation extraction corpus [33], we introduce “ForcedPrompt” as Prompt #2 in Algorithm 1. This prompt adjusts the model’s output to match the ground truth. If there’s a discrepancy between the LLM output and the ground truth, we modify “ForcedPrompt” to generate accurate results, thereby achieving 100% alignment. The revised outputs are then appended to a new word grouping file. The terms ‘natural’ and ‘forced’ are used for their intuitive meanings and have no relation to their usage in electrical circuit theory.

While a forced response prompt can achieve 100% accuracy in condition checks, the unsupervised generation of HRW and LRW groups may introduce bias. To mitigate this, our study employs a dual-model approach using GPT-3.5 and GPT-4, rather than relying on a single Language Model. We enhance the signal-to-noise ratio

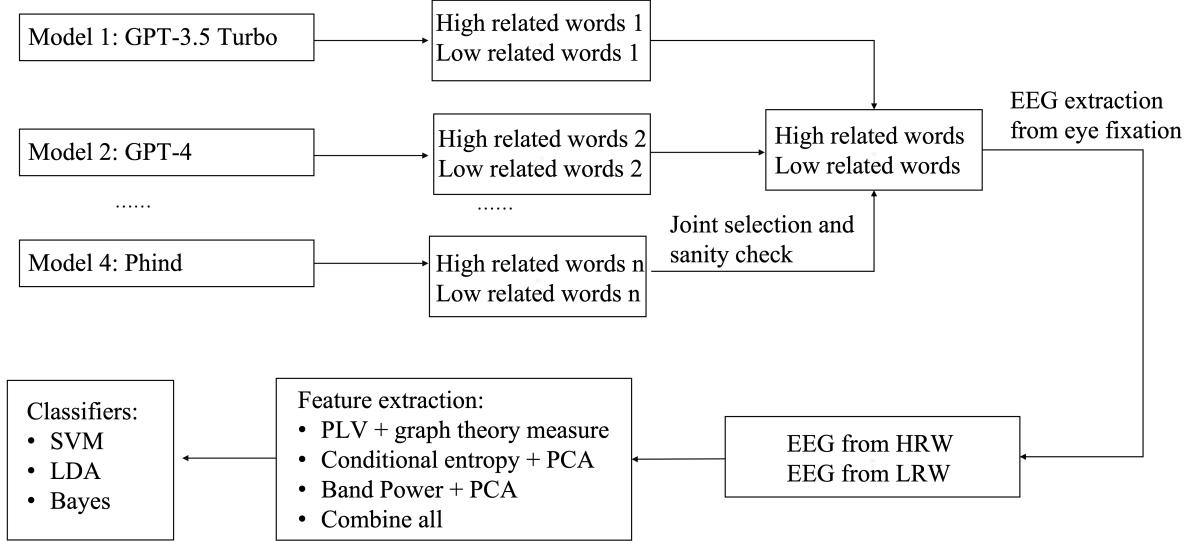


Fig. 2. Binary classification pipeline. This diagram depicts a comprehensive pipeline for analyzing EEG signals from study participants. Initially, two language models evaluate sentences and classify words as either ‘High’ or ‘Low’ relevance. Subsequently, a joint selection process identifies a shared set of HRWs. Leveraging eye-gaze data from the subjects, we extract corresponding EEG signals. We use four distinct feature-extraction techniques to condense information from these signals, reducing their complexity. Finally, these refined features are fed into three separate classifiers, following a standard procedure in brain-computer interface pipelines, to perform binary HRW/LRW classification.

within the HRW-LRW dataset through a joint selection process across all generated datasets, i.e., we select words that belong to both groups.

B. Physiological data processing

1) *Pipeline overview* : Fig. 2 depicts the overview of neural and physiological data processing pipelines. After the joint selection of the HRW and LRW word groups, we extract the eye fixations and fixation-locked EEG data for binary classification tasks. To improve the signal-to-noise ratio (SNR), we employed three feature extraction methods across domains of time-frequency analysis, information theory, connectivity network, and their combined features; these will be elaborated in subsequent sections. An embedded classifier architecture was utilized, incorporating established classifiers such as Support Vector Machine (SVM) and Discriminant Analysis. For Fixation-Related Potential (FRP) analysis, EEG signal extraction was restricted to a predefined time window for each word, ranging from 100ms pre-fixation to 400ms post-fixation.

2) *FRP Analysis*: In contrast to one-dimensional ERP averages, which can obscure dynamic information and inter-trial variability [34], we employed ERPIimage for a two-dimensional representation that allows for trial-by-trial analysis. Utilizing the ERPIimage.m function in the eeglab toolbox (MATLAB 2022b, EEGlab 2020), we generated FRPs for both HRWs and LRWs across 12 subjects. A smoothing parameter of 10 was applied

to enhance the clarity of the FRPimage, which span a temporal window from 100ms pre-fixation to 400ms post-fixation, resulting in a comprehensive ERP signal duration of 500ms.

3) *EEG feature extraction: Band power*: We calculated the power in five EEG frequency bands: delta (0.5-4 Hz), theta (4-8 Hz), alpha (8-13 Hz), beta (13-30 Hz), and gamma (30-64 Hz). We employed MATLAB’s “bandpower” function from the Signal Processing Toolbox. The band power (BP) $P_{a,b}$ is computed as follows:

$$P_{a,b} = \int_a^b P(\omega)d\omega = \int_a^b |F(\omega)|^2 d\omega \quad (1)$$

Where $P_{a,b}$ represents the power in the frequency band $[a, b]$, $P(\omega)$ denotes the power spectral density, $|F(\omega)|^2$ is the squared magnitude of the Fourier transforms, with a and b being the lower and upper bounds of the frequency band, respectively. The EEG data comprised 105 channels, resulting in 525 feature variables per trial. To address the challenge posed by this extensive variable set, many of which exhibited redundancy, we used Principal Component Analysis (PCA) to reduce the dimensionality of the data to 30 variables.

Conditional entropy: This study used conditional entropy (CondEn) to extract features of the EEG trail. It serves as a metric quantifying the level of mutual information between the two random variables. The mutual information between two discrete random variables is

defined as follows:

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)} \right) \quad (2)$$

Where $p(x)$ is the approximate density function. By employing this approach, the mutual information $I(X;Y)$ is computed, establishing its connection with the CondEn $I(X;Y)$.

$$H(X|Y) = - \sum_{y \in Y} p(y) \sum_{x \in X} p(x|y) \log_2 p(x|y) \quad (3)$$

Where $H(X|Y)$ is the CondEn of X given Y , $p(y)$ is the probability of occurrence of a value y from Y , $p(x|y)$ is the conditional probability of x given y , the sums are performed over all possible values of x in X and y in Y . For 105 EEG channels, we generate a 105-by-105 CondEn matrix. This matrix is asymmetric because mutual information and CondEn measure different aspects of the relationship between X and Y . Flattening this matrix results in over 10,000 feature variables. To manage this high dimensionality, we focus on one half of the matrix and apply PCA to reduce the feature space to 30 principal components.

Connectivity network: The human brain is an expansive and intricate network of electrical activity akin to a vast ocean of electric currents [35]. Understanding the intricate connections within the brain and quantifying its connectivity has garnered increasing interest [36]–[38]. This study employed the Phase Locking Value (PLV) to construct a weighted undirected brain connectivity network [33]. Each channel is represented as a node in the graph, and we depict the correlation strength between channels as the edges connecting them.

After constructing the weighted brain network, a range of graph theory measurements can be used as features for analyzing EEG signals. These measurements capture various aspects of the network's structure and organization, including degree, similarity, assortativity, and core structures [39], [40]. We use the clustering coefficient to reduce the dimension to 30 variables.

$$C(v) = \frac{2e(N(v))}{|N(v)|(|N(v)| - 1)} \quad (4)$$

In this equation, $2e(N(v))$ counts the total number of edges in the neighborhood of v , and $|N(v)|(|N(v)| - 1)$ is the total number of possible edges in the neighborhood of v . The coefficient 2 in the numerator accounts for each edge connecting two vertices and is counted twice. The clustering coefficient provides insights into the tendency of nodes in a graph to form clusters or communities, with higher values indicating a greater density of interconnected nodes [40].

Combine all three features: Inspired by [41], combining features from different domains might improve the quality of features and classification performance. We concatenate the three features we introduced above, resulting in 90 variables.

4) Machine learning classifiers and feature selection: Initially, the features—BP, CondEn, and PLV-connectivity network—have high dimensions with original dimensions of 525 (105×5), 5565, and 5565 ($\frac{(11025 - 105)}{2} + 105$), respectively. We reduced the input variables for subsequent classifier training to 30 for each feature by applying PCA and the clustering coefficient for feature selection. Generally, Discriminant Analysis and SVMs are frequently used as non-neural network classifiers in BCI [42]. We incorporated features extracted from EEG signals to train 11 classifiers simultaneously: LDA, QDA, Logistic Regression, Gaussian Naive Bayes, Kernel Naive Bayes, Linear SVM, Quadratic SVM, Cubic SVM, Fine Gaussian SVM, Medium Gaussian SVM, and Coarse Gaussian SVM. The highest classification accuracy is selected as the final result. To ensure the validity of our outcomes, particularly for smaller sample groups, we report 5-fold cross-validation accuracy.

Given the significant class imbalance—LRW EEG data points outnumbering HRW by over 3:1—we applied non-repetitive random downsampling to the LRW class. This ensures equal representation of HRW and LRW data points in the training set. Consequently, the chance label of validation accuracy is 50%.

While deep learning approaches like EEGnet have shown promise in EEG classification [43], [44], their core feature extraction layers are primarily designed for image data [45]. The applicability of such methods to time-series EEG data remains a subject of ongoing discussion. We refrained from using deep neural network techniques in this study to maintain model explainability.

IV. RESULTS

This section presents the results of our pipeline. We first present the results concerning the LLM comparisons, providing statistical insights into the distinctions between GPT-3.5 and GPT-4. Then, we delve into the specifics of each relation class, aiming to gain a more profound understanding. Then, we demonstrate eye fixation statistics for HRWs and LRWs. Next, we highlight the ERP analysis of the Fixation-locked EEG signal. Finally, we present the results of our binary classification.

A. LLM result analysis

1) GPT-3.5 and GPT-4 comparison: During our experimental investigation involving a state-of-the-art large language model, we observed a remarkable level of

TABLE I
MODEL ACCURACY FOR TASK 1 AND TASK3

	12 subjects	GPT-3.5 Turbo	GPT-4	LLaMA	Phind
Task 1	79.53 ± 11.22	93.74 ± 1.99	97.44 ± 0.83	95.17 ± 2.13	96.07 ± 1.73
Task 3	93.16 ± 4.93	95.59 ± 1.48	98.82 ± 0.94	95.80 ± 2.16	97.14 ± 1.28

accuracy when the model was tasked with answering reading comprehension questions from Tasks 1 and 3. Table I compares the performance of different language models on ZuCo Task 3 with that of 12 subjects. Given large language models’ generative and non-deterministic nature, each experimental run produced slightly varying outputs. To mitigate this variability and optimize resource utilization, we executed each model five times and calculated the mean of their responses as the final output. As we can see from Table I, GPT-4 has the highest mean and lowest standard deviation among 12 subjects and all four LLMs over Tasks 1 and 3. Task 1 focused on sentiment inference, and 12 subjects generally have lower accuracy than Task 3. We didn’t include Task 2 because it shares the same corpus with Task 3. While GPT-3.5 attained a lower score of 95.59%, it still outperformed all subjects.

GPT-3.5 and GPT-4 categorize words into HRW and LRW sets for all sentences in Task 3. Specifically, GPT-3.5 generates the first group of HRW and LRW, while GPT-4 produces the second group. By “joint selection,” we identify common elements between these first and second HRW groups to create a third HRW group, leaving the remaining words to constitute the third LRW group. Unless otherwise stated, references to HRWs and LRWs refer to the third group, jointly selected by GPT-3.5 and GPT-4.

B. Eye-fixation statistics

Next, we analyzed the eye activities during the reading process. Table II compares the fixation counts and five additional eye-fixation features for HRWs and LRWs. We excluded the “VISITED” category from the initial nine categories of relationships, resulting in 7271 words distributed among the remaining eight categories after the commonset selection of GPT-3.5 and GPT4. Among these eight categories, LRWs significantly outnumbered HRWs by a six-to-one ratio, with 6,109 LRWs and 1,162 HRWs. Subsequently, we analyzed the fixation per word metric for the HRW and LRW categories for all 12 subjects. Note that the data from three subjects were incomplete for one or two relationships. Table II shows that HRWs received an average of 1.0584 fixations per word, while LRWs received 0.6576 fixations per word. We performed these calculations both with and without

considering zero-fixation words. We presented the results in the second and third columns of the table.

In our analysis, we also considered excluding words that received no fixations, followed by comparing average fixation counts between two distinct categories: HRWs and LRWs. The eye-fixation comparison between no-fixation word excluded and included is shown in Fig. 3 for all 12 subjects. We undertook this step because words lacking any fixations are predominantly associated with the LRW category. Our results show HRWs had an average of slightly more fixations per word than LRWs, with values of 1.5126 and 1.4026, respectively. This discovery aligns with our initial expectations, rooted in the dilution effect of the larger number of LRWs.

We also compared five eye-fixation features, as presented in the last five columns of Table II. Generally, these features all measure the duration of a reader’s gaze on a word, capturing nuances of first-pass reading, regressions and distinguishing between one or multiple fixations. Among these eye-fixation features, HRWs exhibited higher values than LRWs for four out of five metrics, except for SFD. Furthermore, four out of five features showed statistically significant differences, except for the GPT.

C. Fixation-related potentials

The subsequent analysis illustrates the FRP for nine subjects. We excluded three additional subjects because of incomplete data regarding at least one keyword relationship.

Fig. 4 shows the ERPimage time-locked to fixation onsets for HRWs and LRWs for Subject ZAB, complemented by the mean FRP and power spectral density. Power spectral density for the two (HRW and LRW) conditions demonstrates the most significant differences within the [0.5, 10] Hz and [25, 45] Hz ranges, indicative of delta and gamma band activities.

Fig. 5 shows the topographic maps representing the average band power across five frequency bands for nine subjects. We excluded three subjects because of missing data in one or two relations within the total set of eight relations. The topographic maps in the first and second rows correspond to HRWs and LRWs. The third row displays the differential BP between HRWs and LRWs. Across all frequency bands, we observe a

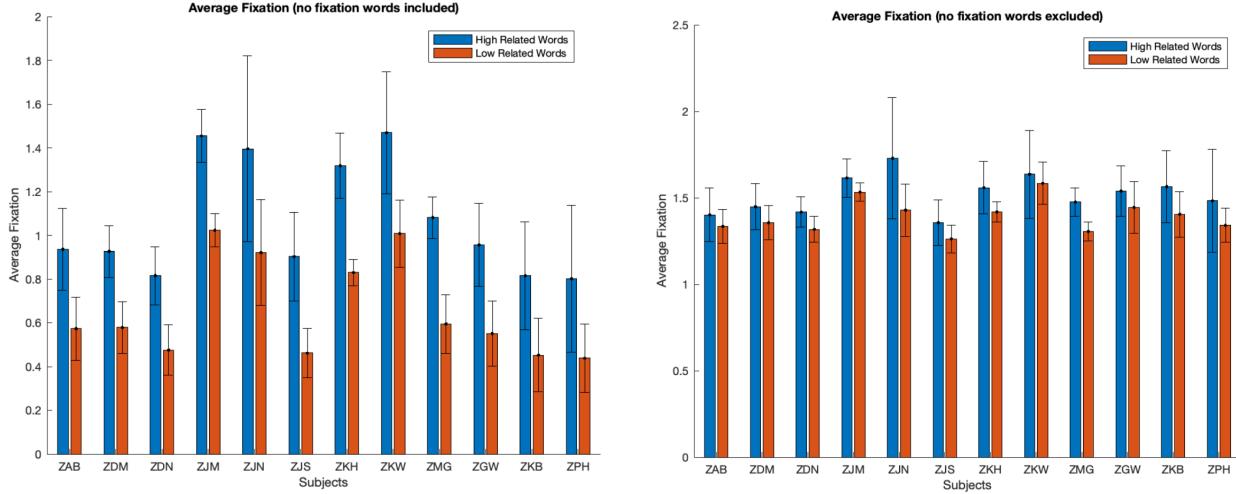


Fig. 3. **Average fixation counts on the HRWs and LRWs.** The left figure displays the average fixation count across 12 subjects, including words without receiving any fixations. “No-fixation” words appear in both HRW and LRW groups. The average fixation count for HRWs appears much greater in this plot. In contrast, the right figure presents the same comparison but excludes words with no fixations, providing a more robust assessment of the average fixation differences between HRW and LRW. As expected, when we omit instances of no-fixation words, the average fixation count for LRWs increases significantly. However, it’s noteworthy that even with this adjustment, the average fixation count for HRWs remains higher than that of LRWs across all subjects. This observation supports the hypothesis that subjects focus more on words closely aligned with the keyword. The whiskers in the figures represent the standard deviation across the eight keyword relations.

significant concentration of power primarily localized in occipital scalp regions, particularly within the delta and theta bands. This localization reflects the involvement of visual word-processing mechanisms. It’s plausible to suggest that related and unrelated words initiate distinct perceptual processes, which could be attributed to top-down attentional modulation [46], [47]. Nevertheless, the most salient differences in BP are within the delta and gamma bands. These disparities may be linked to neural mechanisms that underlie semantic integration and comprehension, as discussed in [48].

D. Binary classification analysis

1) *Subject-wise classification results:* This study assessed the viability of using fixation-locked EEG data to detect whether participants looked at HRWs or LRWs. As previously mentioned, we determined the relevance labels using the GPT-3.5 and GPT-4 models and reported the highest validation accuracies of eleven classifiers. Fig. 6 visually represents the number of HRW and LRW samples reported by the GPT-3 and GPT-4 models and the overlapping data they share across twelve subjects. Each subject exhibited distinct reading patterns, and some, such as ZJM, ZJN, ZKH, and ZKW, showed notably high eye fixations per word. Consequently, this group of subjects contributed more EEG training data.

First, we explored the differences between using word labels generated by different LLMs. We employed a 5-fold cross-validation approach for HRW versus LRW

classification. Fig. 7 illustrates the classification accuracy of words labeled by GPT-3.5, GPT-4, and words jointly labeled by both LLMs, based on Linear SVM. Notably, among the three LLM-based methods for HRW and LRW grouping, the common HRW selection achieved the highest mean accuracy. Importantly, all mean classification accuracies surpass the chance level by jointly labeled data.

Upon scrutinizing the average validation accuracy across the spectrum of the GPT models for each respective subject, it was discernible that an enhanced performance was typically recorded when the GPT-3.5 and GPT-4 models were employed in conjunction, as opposed to using either the GPT-3.5 or GPT-4 model in isolation.

Next, we delve into the detailed comparisons of classification accuracy when we used four different features as inputs to 11 machine-learning classifiers. Fig. 8 shows the classification accuracy of words jointly labeled by both LLMs. This figure compares classification performance based on different EEG features. The “combine” and “CondEn” methods consistently have the highest validation accuracy across most subjects.

In an individual subject context, we found that the Subjects ZDM, ZDN, ZJN, and ZKW characteristically showed superior validation accuracy across all the feature extraction methodologies and iterations of the GPT models. This could show more consistency in the EEG classification within their data sets. Conversely, Subjects

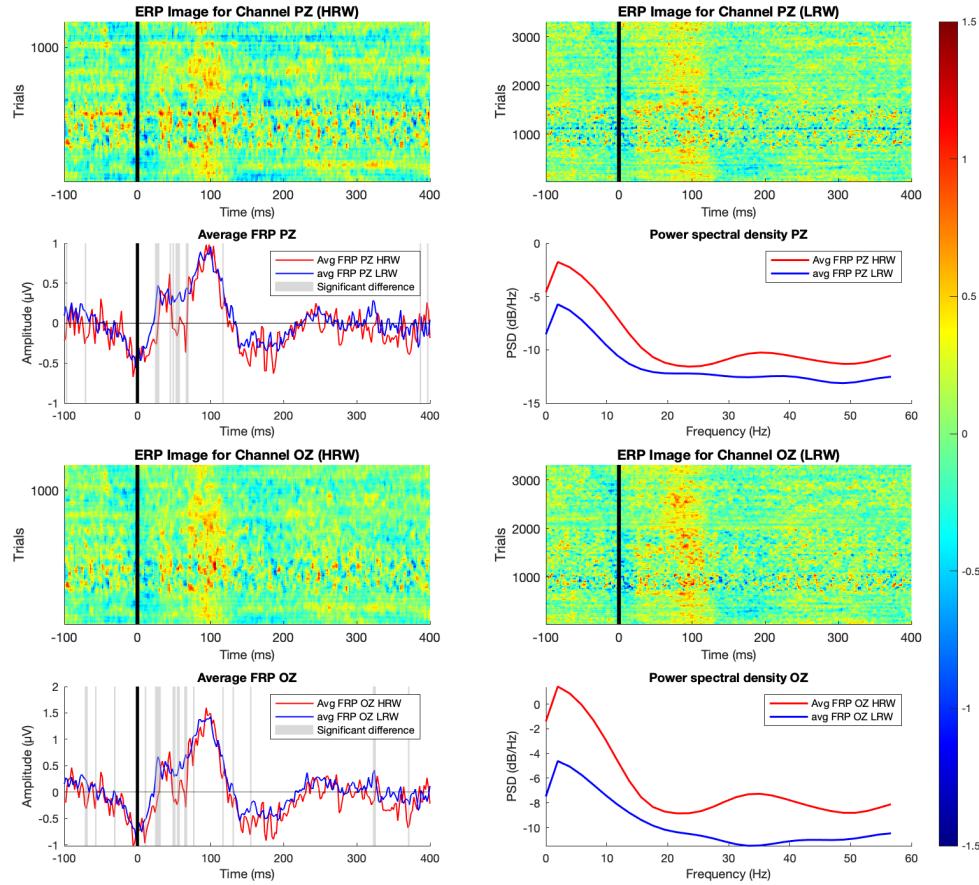


Fig. 4. FRP and Power Spectral Density Analysis for Subject ZAB in HRW and LRW Conditions. The figure presents ERPinages for channels Pz and Oz for both groups (HRW and LRW). Accompanying the ERPinages are mean FRPs and power spectral densities for both conditions across the channels. Areas of significant difference in the FRPs are highlighted in green. Notable disparities in power spectral density occur within the [0.5, 10] Hz and [25, 45] Hz frequency ranges, corresponding to delta and gamma band activities.

ZGW, ZKB, and ZPH typically showed a diminished average validation accuracy.

2) *Classifier performance analysis:* We thoroughly investigated the efficacy of several machine-learning classifiers when applied to words labeled jointly by GPT-3.5 and GPT-4, as delineated in Fig. 8. Four distinct feature sets served as inputs for evaluating these classifiers. The first set amalgamates all three techniques, as seen in Fig. 8(A), while the second set intertwines BP with PCA, as referenced in Fig. 8(B). The third set fuses CondEn with PCA, illustrated in Fig. 8(C), and the final set pairs PLV with the clustering coefficient, demonstrated in Fig. 8(D). Notably, linear classifiers achieved the highest accuracy, reaching 62.1% (on Subject ZPH).

Fig. 8 provides a comprehensive view of the classification accuracy results, whereas Table III summarizes the average and standard deviation of classification performance among 12 subjects, using four different feature sets and eleven machine-learning algorithms. We noted a

tangible variation in the accuracy of the classifiers across distinct methodologies and subjects in the Table. The Linear SVM consistently outperformed other algorithms, exhibiting peak accuracy of $60.03 \pm 1.72\%$ in combined features scenarios. Using the second feature set (BP + PCA) resulted in a marginal decrement in the accuracy of all classifiers, with the highest recorded at $56.73 \pm 1.80\%$ using Medium Gaussian SVM. In contrast, the third set (CondEn + PCA) enhanced accuracy for specific classifiers, with the Linear SVM being paramount, achieving $59.37 \pm 2.05\%$ at its highest. Conversely, employing the fourth set (PLV + clustering coefficient) precipitated a universal decline in overall accuracy across all classifiers, pinpointing $54.70 \pm 2.80\%$ for Linear SVM.

V. DISCUSSION AND CONCLUSION

This pilot study introduced a novel BCI pipeline that synergistically combines LLMs, particularly Generative Pre-trained Transformers (GPT-3.5 and GPT-4), and an

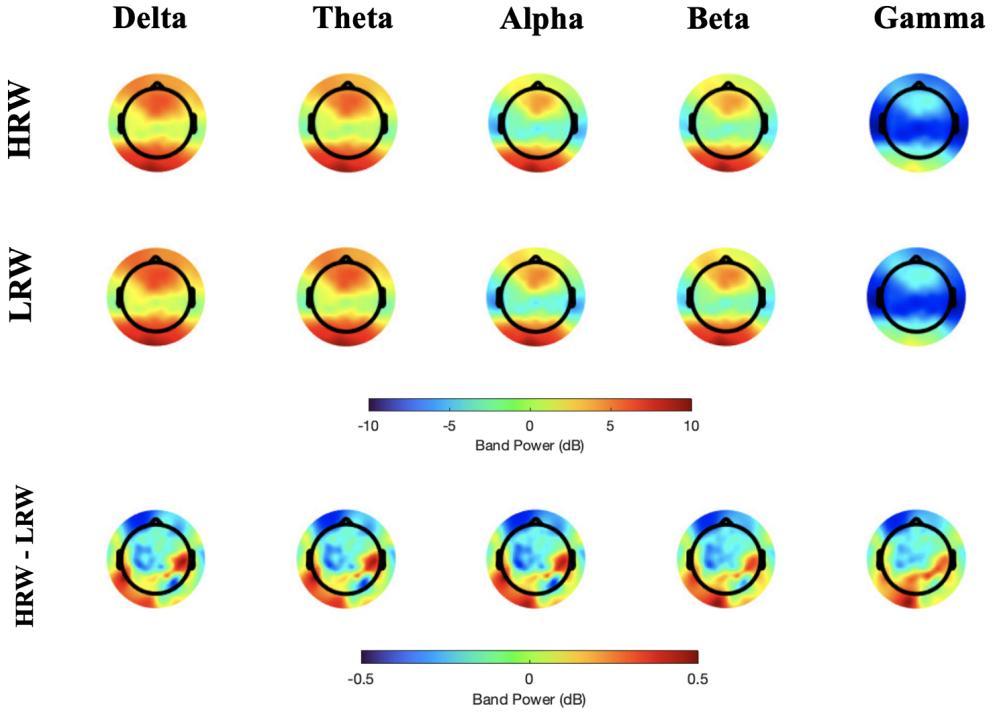


Fig. 5. Topographic Maps of BP Across Five Frequency Bands. The figure depicts the average band power for nine subjects, excluding three due to missing data. The first and second rows show topographic maps for HRWs and LRWs, respectively, while the third row illustrates the differential BP between the two groups. Across all frequency bands, power is significantly concentrated in the occipital scalp regions, especially within the delta and theta bands, suggesting the role of visual word processing mechanisms. Notably, the most distinct differences in band power are observed in the delta and gamma bands, which may relate to neural mechanisms involved in semantic integration and comprehension.

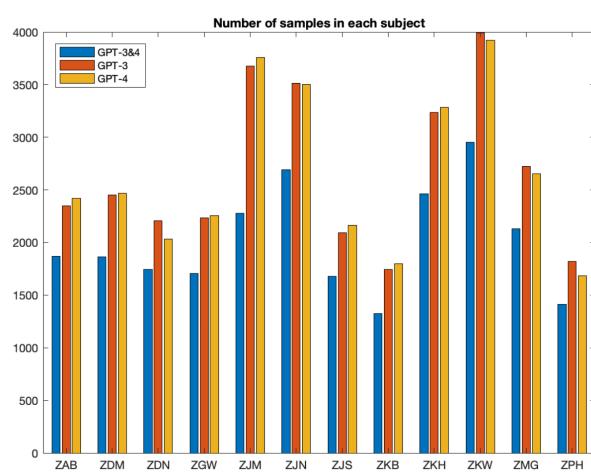


Fig. 6. EEG epoch counts for twelve subjects. The figure displays the numbers of HRW and LRW samples for twelve subjects. Subjects with distinct reading patterns, specifically ZJM, ZJN, ZKH, and ZKW, exhibited high eye fixations per word and thus contributed more EEG training data. The graph highlights the trade-off between word accuracy and the volume of data points crucial for machine-learning classification.

EEG-based BCI. This is one of the first efforts to use GPT capability for this specialized intersection of neuroscience and artificial intelligence.

Eye gaze is a prominent biomarker, holding crucial information for comprehending cognitive processes in individuals involved in task-specific reading activities [49]. In this study, we conducted average fixation analyses across three distinct dimensions: on a subject-by-subject basis, concerning specific semantic relations, and at the level of individual words. We performed these analyses on data collected from 12 participants and encompassing eight different semantic relations. Our results unequivocally show that participants allocate significantly more time to words that exhibit high semantic relevance to specific relations (i.e., keywords) during inference tasks. Appendices A and B provide additional support for this observation.

Unlike traditional BCIs, which relied on precise stimulus presentation as timing markers to extract event-related EEG activities such as P300 and Steady-State Visual Evoke Potentials in well-controlled laboratory environments, our approach leveraged fixation onsets to

TABLE II
EYE-FIXATION STATISTICS

	# Word count (per subject)	# Fixation (no fixation words included)	# Fixation (no fixation words excluded)	Gaze duration (GD)
High RW	1162	1.0584 ± 0.2721	1.5126 ± 0.1134	133.1522 ± 23.2412
Low RW	6109	0.6576 ± 0.2278	1.4026 ± 0.0967	124.8666 ± 22.3508
Total Sample Size	7271	-	-	-
P-value	-	7.4666e-4	1.7902e-2	2.1496e-11

	Total reading time (TRT)	First fixation duration (FFD)	Single fixation duration (SFD)	Go-past time (GPT)
High RW	183.7525 ± 37.41	113.0653 ± 14.1043	71.5562 ± 5.5873	209.2344 ± 39.6288
Low RW	160.0450 ± 27.1377	110.6034 ± 14.4297	79.5498 ± 7.9179	206.9365 ± 33.0659
Total Sample Size	-	-	-	-
P-value	3.4834e-4	1.323e-4	4.4111e-5	0.06493

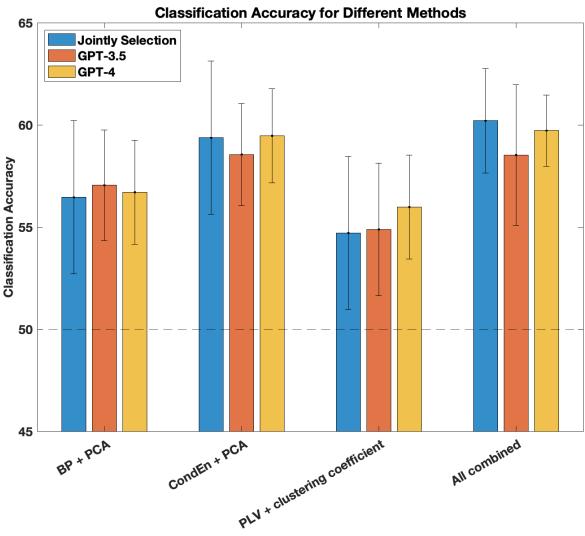


Fig. 7. A comparison of classification accuracy on words labeled as HRWs and LRWs by various LLMs. Classification performance, based on Linear SVM, was evaluated considering three LLM-based word selections and four feature-extraction methods, with the EEG feature CondEn exhibiting superior performance. A combination of all three EEG features rendered the highest overall performance. Crucially, a marginal enhancement in classification accuracy was observed when identifying HRWs co-selected by GPT-3.5 and GPT-4.

capture EEG signals related to words during natural reading. This implementation significantly enhances the practicality of BCIs for real-world applications.

We evaluated the performance of four distinct LLMs to improve classification outcomes. Our hybrid architecture, combining GPT-3.5 and GPT-4 as word labelers with eye tracking and BCI components, demonstrated remarkable performance, achieving an impressive accuracy rate exceeding 60% in the classification of word relevance. This enhancement was realized by applying SVMs to three domain-specific features: BP, CondEn combined with PCA, and PLV-based graph theory techniques. Carefully chose each feature for its well-established utility in BCI research and its capacity to enhance the signal-to-noise ratio. Additionally, we explored the pair-wise coherence of 5-frequency bands but ultimately decided against its use because of its computational complexity, particularly when considering the 105 EEG channels we employed.

Furthermore, we comprehensively analyzed single-word fixation statistics for 12 subjects, encompassing eight classes within the HRW and LRW groups. To account for the absence of data in eight relationship instances — Subject ZGW did not include “JOB”, ZKB lacked “WIFE,” and ZPH lacked both “POL AFF” and “WIFE”—we ultimately generated 184 figures ($12 \times 8 \times 2 - 8$), all of which are included in the supplementary materials. Our findings revealed that words within

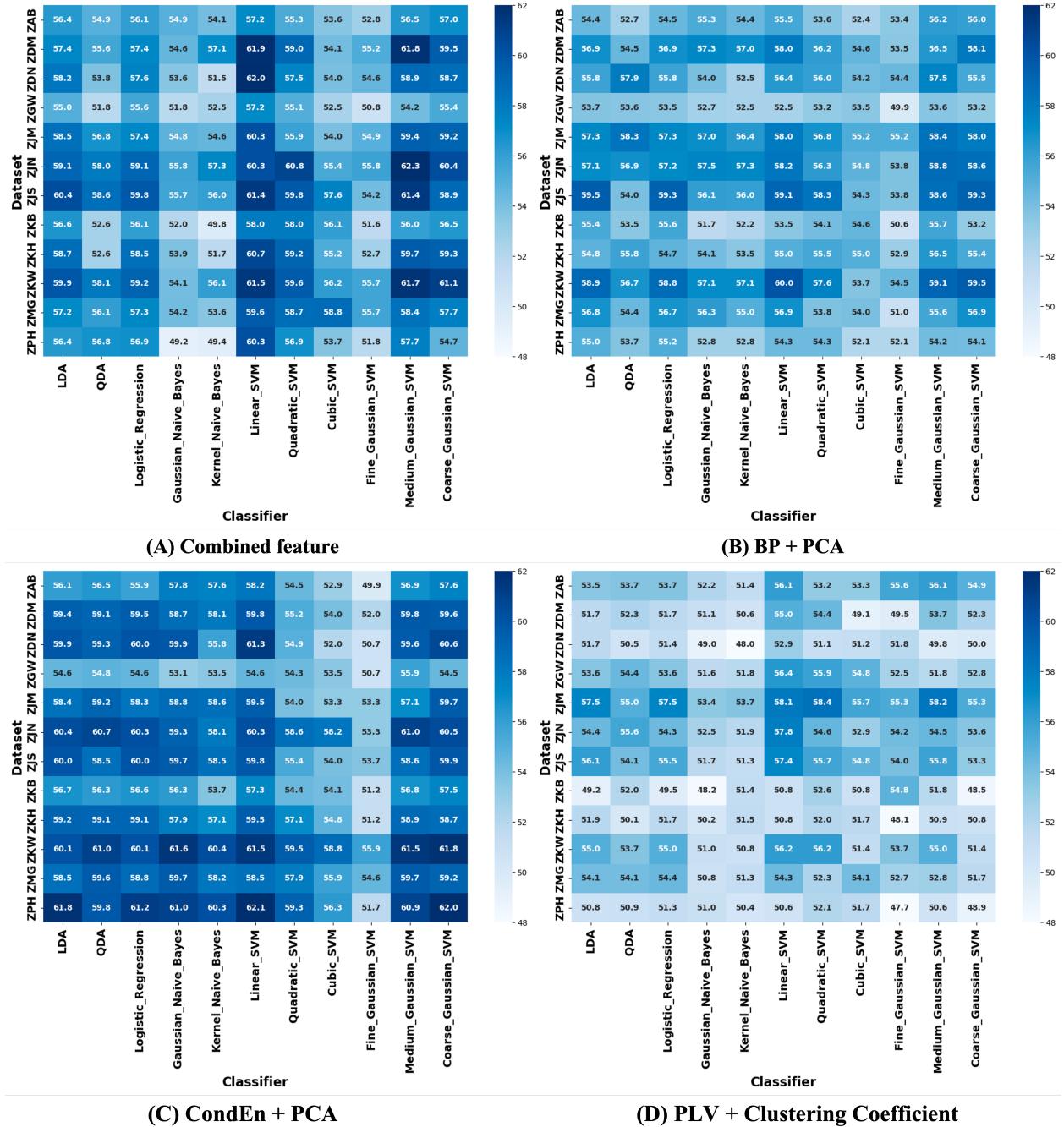


Fig. 8. Comparison of Different Classifiers. This figure depicts the classification accuracy of several machine-learning classifiers applied to four input feature sets. Classifiers include LDA, QDA, Logistic Regression, Gaussian and Kernel Naive Bayes, and several types of SVMs, totaling 11 classifiers. The feature sets include (A) combined features, (B) BP + PCA, (C) CondEn + PCA, and (D) PLV + clustering coefficient. Overall, SVM and its variants outperform the other classifiers. The highest accuracy achieved using linear classifiers reaches up to 62.1% (Subject ZPH).

TABLE III
MEAN ACCURACY \pm STANDARD DEVIATION ACROSS SUBJECTS

	Combined	BP+PCA	ConEn+PCA	PLV+Clustering Coef.
LDA	57.82 \pm 1.60	56.30 \pm 1.77	58.76 \pm 2.04	53.29 \pm 2.34
QDA	55.48 \pm 2.34	55.17 \pm 1.88	58.66 \pm 1.86	53.03 \pm 1.83
Logistic Regression	57.58 \pm 1.34	56.29 \pm 1.74	58.70 \pm 2.00	53.30 \pm 2.24
Gaussian Naive Bayes	53.72 \pm 1.88	55.16 \pm 2.03	58.65 \pm 2.26	51.06 \pm 1.44
Kernel Naive Bayes	53.64 \pm 2.71	54.73 \pm 1.99	57.49 \pm 2.20	51.18 \pm 1.30
Linear SVM	60.03 \pm 1.72	56.45 \pm 2.33	59.37 \pm 2.05	54.70 \pm 2.80
Quadratic SVM	57.98 \pm 1.86	55.48 \pm 1.67	56.26 \pm 2.09	54.04 \pm 2.19
Cubic SVM	55.10 \pm 1.83	54.03 \pm 0.97	54.82 \pm 2.09	52.63 \pm 1.97
Fine Gaussian SVM	53.82 \pm 1.79	52.93 \pm 1.68	52.35 \pm 1.81	52.49 \pm 2.72
Medium Gaussian SVM	59.00 \pm 2.57	56.73 \pm 1.80	58.89 \pm 1.85	53.42 \pm 2.57
Coarse Gaussian SVM	58.20 \pm 1.97	56.48 \pm 2.26	59.30 \pm 2.06	51.96 \pm 2.17

the HRW group garnered significantly higher average fixation counts than those in the LRW group. These findings provide valuable insights into how participants comprehend the reading corpus.

Despite these advances, the study has several limitations. This study faces challenges because of the ‘black box’ nature of LLMs, particularly in the context of the non-deterministic relation, such as ‘AWARD,’ where certain outputted words appear incongruous. This limitation might affect our findings’ generalizability and underscore the need for a quantitative assessment to ensure the accuracy and validity of keyword identification.

Additionally, contextual complexities often influence semantic classifications. For example, “gold” acquire distinct semantic relevance when juxtaposed with terms like “medal.” The sentences incorporating specific target terms, such as “NATIONALITY” or “WIFE,” exhibit a significant disparity in the distribution between HRW and LRW, making them more deterministic. These discrepancies add complexity to the classification of EEG data and introduce the possibility of contamination within the dataset, especially when the meaning of words is most effectively comprehended within the context of phrases rather than in isolation.

This study underscores the potential for more expansive research on elucidating reading-related cognitive behaviors. The promise of integrating LLMs into BCIs also points towards future advancements in reading assistance technologies. While acknowledging its limitations and complexities, our work is an early yet significant contribution, paving the way for more integrated studies to foster a deeper understanding of the multifaceted interplay between neuroscience and computational linguistics.

REFERENCES

- [1] H. Wang, T. Fu, Y. Du, W. Gao, K. Huang, Z. Liu, P. Chandak, S. Liu, P. Van Katwyk, A. Deac *et al.*, “Scientific discovery in the age of artificial intelligence,” *Nature*, vol. 620, no. 7972, pp. 47–60, 2023.
- [2] K. Singhal, S. Azizi, T. Tu, S. S. Mahdavi, J. Wei, H. W. Chung, N. Scales, A. Tanwani, H. Cole-Lewis, S. Pfohl *et al.*, “Large language models encode clinical knowledge,” *Nature*, pp. 1–9, 2023.
- [3] M. Abdullah, A. Madain, and Y. Jararweh, “Chatgpt: Fundamentals, applications and social impacts,” in *2022 Ninth International Conference on Social Networks Analysis, Management and Security (SNAMS)*. IEEE, 2022, pp. 1–8.
- [4] S. Bubeck, V. Chandrasekaran, R. Eldan, J. Gehrke, E. Horvitz, E. Kamar, P. Lee, Y. T. Lee, Y. Li, S. Lundberg *et al.*, “Sparks of artificial general intelligence: Early experiments with gpt-4,” *arXiv preprint arXiv:2303.12712*, 2023.
- [5] D. Gunning, M. Stefk, J. Choi, T. Miller, S. Stumpf, and G.-Z. Yang, “Xai—explainable artificial intelligence,” *Science robotics*, vol. 4, no. 37, p. eaay7120, 2019.
- [6] M. A. Just and P. A. Carpenter, “A theory of reading: from eye fixations to comprehension.” *Psychological review*, vol. 87, no. 4, p. 329, 1980.
- [7] K. Rayner, “Eye movements in reading and information processing: 20 years of research.” *Psychological bulletin*, vol. 124, no. 3, p. 372, 1998.
- [8] W. Kintsch, *Comprehension: A paradigm for cognition*. Cambridge university press, 1998.
- [9] M. Binz and E. Schulz, “Using cognitive psychology to understand gpt-3,” *Proceedings of the National Academy of Sciences*, vol. 120, no. 6, p. e2218523120, 2023.
- [10] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, “Training language models to follow instructions with human feedback,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 27730–27744, 2022.
- [11] C. Pandarinath, P. Nuyujukian, C. H. Blabe, B. L. Sorice, J. Saab, F. R. Willett, L. R. Hochberg, K. V. Shenoy, and J. M. Henderson, “High performance communication by people with paralysis using an intracortical brain-computer interface,” *Elife*, vol. 6, p. e18554, 2017.
- [12] D. Shawky and A. Badawi, “Towards a personalized learning experience using reinforcement learning,” *Machine learning paradigms: Theory and application*, pp. 169–187, 2019.
- [13] M. Kutas and K. D. Federmeier, “Thirty years and counting: finding meaning in the n400 component of the event-related brain potential (erp),” *Annual review of psychology*, vol. 62, pp. 621–647, 2011.
- [14] M. Kutas and S. A. Hillyard, “Reading senseless sentences: Brain potentials reflect semantic incongruity,” *Science*, vol. 207, no. 4427, pp. 203–205, 1980.
- [15] W. Kintsch, “The role of knowledge in discourse comprehension: a construction-integration model.” *Psychological review*, vol. 95, no. 2, p. 163, 1988.
- [16] J. S. B. Evans, “Dual-processing accounts of reasoning, judgment, and social cognition,” *Annu. Rev. Psychol.*, vol. 59, pp. 255–278, 2008.

- [17] D. E. Rumelhart, "Schemata: The building blocks of cognition," in *Theoretical issues in reading comprehension*. Routledge, 2017, pp. 33–58.
- [18] R. C. Anderson, "Role of the reader's schema in comprehension, learning, and memory," in *Theoretical models and processes of literacy*. Routledge, 2018, pp. 136–145.
- [19] P. N. Johnson-Laird, *Mental models: Towards a cognitive science of language, inference, and consciousness*. Harvard University Press, 1983, no. 6.
- [20] D. S. McNamara and J. Magliano, "Toward a comprehensive model of comprehension," *Psychology of learning and motivation*, vol. 51, pp. 297–384, 2009.
- [21] L. Bareta, L. M. B. Tomitch, V. K. Lim, and K. E. Waldie, "Investigating reading comprehension through eeg," *Ilha do Desterro: A Journal of English Language, Literatures in English and Cultural Studies*, no. 63, pp. 69–99, 2012.
- [22] M. H. Bornstein and M. E. Lamb, *Cognitive development: An advanced textbook*. Taylor & Francis, 2011.
- [23] M. F. Mridha, S. C. Das, M. M. Kabir, A. A. Lima, M. R. Islam, and Y. Watanabe, "Brain-computer interface: Advancement and challenges," *Sensors*, vol. 21, no. 17, p. 5746, 2021.
- [24] Q. Li, *Reading Comprehension Analysis and Prediction Based on EEG and Eye-Tracking Techniques*. University of California, San Diego, 2021.
- [25] H. Zeng, C. Yang, G. Dai, F. Qin, J. Zhang, and W. Kong, "Eeg classification of driver mental states by deep learning," *Cognitive neurodynamics*, vol. 12, pp. 597–606, 2018.
- [26] R. J. Seitz, R. Schäfer, D. Scherfeld, S. Friederichs, K. Popp, H.-J. Wittsack, N. Azari, and M. Franz, "Valuating other people's emotional face expression: a combined functional magnetic resonance imaging and electroencephalography study," *Neuroscience*, vol. 152, no. 3, pp. 713–722, 2008.
- [27] M. Tanaka, A. Ishii, and Y. Watanabe, "Neural effects of mental fatigue caused by continuous attention load: a magnetoencephalography study," *Brain research*, vol. 1561, pp. 60–66, 2014.
- [28] A. C. Jenkins, "Rethinking cognitive load: a default-mode network perspective," *Trends in Cognitive Sciences*, vol. 23, no. 7, pp. 531–533, 2019.
- [29] Q. Wang, S. Yang, M. Liu, Z. Cao, and Q. Ma, "An eye-tracking study of website complexity from cognitive load perspective," *Decision support systems*, vol. 62, pp. 1–10, 2014.
- [30] N. Hollenstein, J. Rotsztejn, M. Troendle, A. Pedroni, C. Zhang, and N. Langer, "Zuco, a simultaneous eeg and eye-tracking resource for natural sentence reading," *Scientific data*, vol. 5, no. 1, pp. 1–13, 2018.
- [31] H. Brouwer, H. Fitz, and J. Hoeks, "Getting real about semantic illusions: Rethinking the functional role of the p600 in language comprehension," *Brain research*, vol. 1446, pp. 127–143, 2012.
- [32] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit," in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 2014, pp. 55–60.
- [33] S. Aydore, D. Pantazis, and R. M. Leahy, "A note on the phase locking value and its properties," *Neuroimage*, vol. 74, pp. 231–244, 2013.
- [34] T.-P. Jung, S. Makeig, M. Westerfield, J. Townsend, E. Courchesne, and T. J. Sejnowski, "Analyzing and visualizing single-trial event-related potentials," in *Advances in Neural Information Processing Systems*, M. Kearns, S. Solla, and D. Cohn, Eds., vol. 11. MIT Press, 1998.
- [35] M. Rubinov and O. Sporns, "Complex network measures of brain connectivity: uses and interpretations," *Neuroimage*, vol. 52, no. 3, pp. 1059–1069, 2010.
- [36] Y. Zhang, Y. Liao, Y. Zhang, and L. Huang, "Emergency braking intention detect system based on k-order propagation number algorithm: a network perspective," *Brain Sciences*, vol. 11, no. 11, p. 1424, 2021.
- [37] W. Ding, Y. Zhang, and L. Huang, "Using a novel functional brain network approach to locate important nodes for working memory tasks," *International journal of environmental research and public health*, vol. 19, no. 6, p. 3564, 2022.
- [38] Y. Chen, Y. Zhang, W. Ding, F. Cui, and L. Huang, "Research on working memory states based on weighted-order propagation number algorithm: An eeg perspective," *Journal of Sensors*, vol. 2022, 2022.
- [39] A. Fornito, A. Zalesky, and E. Bullmore, *Fundamentals of brain network analysis*. Academic press, 2016.
- [40] E. Bullmore and O. Sporns, "Complex brain networks: graph theoretical analysis of structural and functional systems," *Nature reviews neuroscience*, vol. 10, no. 3, pp. 186–198, 2009.
- [41] K.-J. Chiang, S. Dong, C.-K. Cheng, and T.-P. Jung, "Using eeg signals to assess workload during memory retrieval in a real-world scenario," *Journal of Neural Engineering*, vol. 20, no. 3, p. 036010, 2023.
- [42] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger, "A review of classification algorithms for eeg-based brain-computer interfaces: a 10 year update," *Journal of Neural Engineering*, vol. 15, no. 3, p. 031005, apr 2018. [Online]. Available: <https://dx.doi.org/10.1088/1741-2552/aaebf2>
- [43] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, vol. 15, no. 5, p. 056013, jul 2018. [Online]. Available: <https://dx.doi.org/10.1088/1741-2552/aace8c>
- [44] A. Craik, Y. He, and J. L. Contreras-Vidal, "Deep learning for electroencephalogram (eeg) classification tasks: a review," *Journal of Neural Engineering*, vol. 16, no. 3, p. 031001, apr 2019. [Online]. Available: <https://dx.doi.org/10.1088/1741-2552/ab0ab5>
- [45] Y. Li, R. Yin, H. Park, Y. Kim, and P. Panda, "Wearable-based human activity recognition with spatio-temporal spiking neural networks," 2022.
- [46] C. Tallon-Baudry, O. Bertrand, M.-A. Hénaff, J. Isnard, and C. Fischer, "Attention Modulates Gamma-band Oscillations Differently in the Human Lateral Occipital Cortex and Fusiform Gyrus," *Cerebral Cortex*, vol. 15, no. 5, pp. 654–662, 09 2004. [Online]. Available: <https://doi.org/10.1093/cercor/bhh167>
- [47] V. Wyart and C. Tallon-Baudry, "How ongoing fluctuations in human visual cortex predict perceptual awareness: baseline shift versus decision bias," *Journal of Neuroscience*, vol. 29, no. 27, pp. 8715–8725, 2009.
- [48] S. Palva, S. Kulasekhar, M. Hämäläinen, and J. M. Palva, "Localization of cortical phase and amplitude dynamics during visual working memory encoding and retention," *Journal of Neuroscience*, vol. 31, no. 13, pp. 5013–5025, 2011.
- [49] S.-C. Chen, H.-C. She, M.-H. Chuang, J.-Y. Wu, J.-L. Tsai, and T.-P. Jung, "Eye movements predict students' computer-based assessment performance of physics concepts in different presentation modalities," *Computers & Education*, vol. 74, pp. 61–72, 2014.

NEURO-GPT: TOWARDS A FOUNDATION MODEL FOR EEG

*Wenhui Cui¹, Woojae Jeong¹, Philipp Thölke², Takfarinas Medani¹,
Karim Jerbi^{2,3,4}, Anand A. Joshi¹, Richard M. Leahy¹*

¹ Ming Hsieh Department of Electrical and Computer Engineering, University of Southern California,
Los Angeles, CA, USA

² Psychology Department, Université de Montréal, Montreal, QC, Canada

³ Mila (Quebec AI research institute), Montreal, QC, Canada,

⁴ UNIQUE (Quebec Neuro-AI research center), QC, Canada

ABSTRACT

To handle the scarcity and heterogeneity of electroencephalography (EEG) data for Brain-Computer Interface (BCI) tasks, and to harness the power of large publicly available data sets, we propose Neuro-GPT, a foundation model consisting of an EEG encoder and a GPT model. The foundation model is pre-trained on a large-scale data set using a self-supervised task that learns how to reconstruct masked EEG segments. We then fine-tune the model on a motor imagery classification task to validate its performance in a low-data regime (9 subjects). Our experiments demonstrate that applying a foundation model can significantly improve classification performance compared to a model trained from scratch, which provides evidence for the generalizability of the foundation model and its ability to address challenges of data scarcity and heterogeneity in EEG. The code is publicly available at <https://github.com/wenhui0206/NeuroGPT>.

Index Terms— Foundation Model, EEG, GPT, Encoder

1. INTRODUCTION

The limited scale of training data for electroencephalography (EEG) based Brain-Computer Interface (BCI) classification tasks poses challenges to applying deep learning models. These models require a large amount of training data to converge and generalize to unseen testing data. However, individual differences can lead to heterogeneous feature representations across subjects [1], which makes it difficult to generalize the model across subjects. EEG’s high-dimensional nature and limited availability for specific tasks create additional barriers to the convergence of these models.

One common approach is to learn generalizable features from large amounts of data using self-supervised learning and then transfer to the task of interest [2]. Here, we address the question of whether we can train a model on large-scale EEG datasets using a self-supervised task and then transfer the pre-trained knowledge to enhance performance on a downstream

task. Large language models (LLMs) in natural language processing (NLP) tasks have proven extraordinarily successful using this approach. Similar models have also shown remarkable performance in other tasks including image and video generation [3], medical question answering [4], and neural activity analysis [5, 6, 7]. Despite the popularity of LLMs, there have been relatively few attempts to adapt them to EEG data. The work in [8] employed a BERT-inspired [9] approach to pre-train a transformer model on massive EEG data using a contrastive self-supervised task. However, it exhibited limited generalizability to downstream tasks.

Here we aim to lay the groundwork for developing foundation models for EEG. We employ a Generative Pre-trained Transformer (GPT) model [10], which uses a decoder-only transformer architecture and is trained to predict the next masked token given a sequence of tokens as input (auto-regressive training). In text-based tasks, a sentence is broken down into tokens as input units to the model. A token can be a few characters or a word, depending on the language and context. To adapt the GPT model to EEG data, we split the whole time series into fixed-length “chunks”, treating each chunk as a token. An EEG encoder is incorporated to extract representative features from raw EEG data. Neuro-GPT is a foundation model consisting of an EEG encoder to extract spatio-temporal features from EEG data, and a GPT model that uses self-supervision to predict the masked chunks. The foundation model is pre-trained on the TUH EEG dataset [11]. We fine-tune the model on a motor imagery classification task where only 9 subjects are available. Experiments showed that the EEG encoder learns meaningful features that are generalizable to the downstream task.

2. METHODS

In this section we introduce the architecture of Neuro-GPT, the pre-training details, and the fine-tuning strategies. We divide the raw EEG data into fixed-length chunks from which we generate a sequence of tokens corresponding to contiguous

ous data chunks. The GPT model then learns to predict masked tokens. Employing chunks of raw EEG signals directly as input tokens to the GPT model would be problematic. Given the high dimensionality and low signal-to-noise ratio of EEG data [12], predicting raw signals is particularly challenging for the GPT model, and it may not learn meaningful features given the presence of noise. Thus, we introduce an EEG encoder [13] comprising convolutional and transformer layers to extract spatio-temporal features from the raw EEG. We input chunks of EEG into the encoder to generate the embeddings. These embeddings serve as a lower-dimensional and denoised representation of the raw EEG signals, not only simplifying the prediction of the masked chunk for the GPT model but also enhancing its ability to capture informative temporal correlations and patterns. The overall Neuro-GPT pipeline is illustrated in Figure 1.

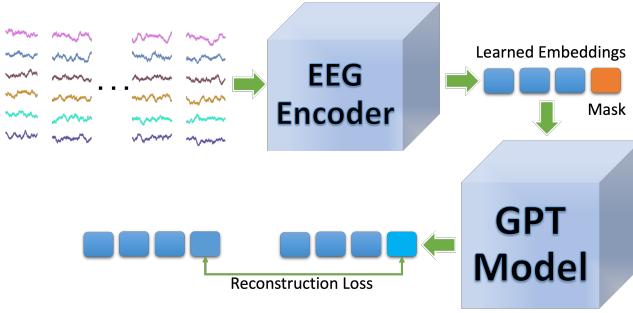


Fig. 1. Neuro-GPT Pipeline: the EEG encoder takes chunks of EEG data as input and generates embeddings as tokens for the GPT model. The last embedded chunk in the sequence is masked. The GPT model then predicts the masked chunk and a reconstruction loss is computed between the prediction and the original embedding token.

2.1. Neuro-GPT Pipeline

EEG Encoder We adopt an encoder architecture incorporating both convolutional and self-attention modules. This arrangement has achieved state-of-the-art performance in BCI classification tasks [13]. We split the raw EEG signals into N chunks, each of time length T . This results in a sequence of chunks denoted $\{D_1, D_2, \dots, D_N\}$. Each chunk is of dimension $C \times T$, where C is the number of channels. Each chunk is treated as an individual training sample in the encoder. In the convolutional module, we apply a temporal convolution filter to the time series and a spatial convolution filter to the electrodes of the EEG. Then after average pooling, the extracted local features are fed into the self-attention layers to incorporate temporal dependencies within a chunk. The self-attention mechanism combined with convolution will encode the spatio-temporal features of each chunk. The outputs of the encoder are the embedded chunks or tokens: $\{\mathcal{H}(D_1), \mathcal{H}(D_2), \dots, \mathcal{H}(D_N)\}$, where \mathcal{H} denotes the map-

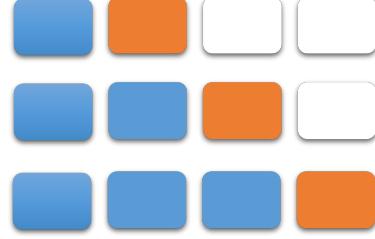


Fig. 2. Causal masking: consider a sequence with four tokens (chunks). We duplicate the sequence three times and progressively mask (represented in orange) one token within each duplicated sequence.

ping learned by the EEG encoder from raw EEG signals to embeddings.

Causal Masking We apply a novel causal masking scheme to the tokens generated by the embedding module. As illustrated in Fig. 2, we first duplicate the sequence of tokens. Starting from the second token, one token is masked and subsequent tokens are zeroed-out in each duplicated sequence. The masked token is replaced with a learnable token \mathcal{M} of the same dimension. So, after causal masking, the input sequence to the GPT model is

$$\begin{aligned} & \{\mathcal{H}(D_1), \mathcal{M}, \mathbf{0}, \dots, \mathbf{0}\}, \\ & \{\mathcal{H}(D_1), \mathcal{H}(D_2), \mathcal{M}, \mathbf{0}, \dots, \mathbf{0}\}, \\ & \dots, \\ & \{\mathcal{H}(D_1), \mathcal{H}(D_2), \mathcal{H}(D_3), \dots, \mathcal{M}\} \end{aligned} \quad (1)$$

The pre-training of Neuro-GPT utilizes a self-supervised task, where the GPT model predicts every masked token in each sequence. We use a causal reconstruction loss defined in Eq. 2 as the self-supervised pre-training objective.

$$\mathcal{L} = \frac{1}{N-1} \sum_{i=2}^N \|\hat{Y}_i - \mathcal{H}(D_i)\|_2^2 \quad (2)$$

$$\text{where } \hat{Y}_i = \mathcal{G}[\mathcal{M} | \mathcal{H}(D_{i-1}), \mathcal{H}(D_{i-2}), \dots, \mathcal{H}(D_1)]$$

where \mathcal{G} denotes the GPT model. We aggregate the reconstruction losses of masked tokens at each position. The predicted token \hat{Y}_i produced by the GPT model is inferred based on the preceding tokens. By predicting the masked token separately from 1, 2, and 3 preceding tokens the model gains insight into the underlying temporal correlations in brain activity across different time scales. Thus the GPT model is potentially able to capture the dynamic evolution of brain activity more accurately.

GPT Model The GPT model employs a decoder-only transformer architecture consisting of a multi-layered stack of self-attention and feed-forward modules, enabling it to capture the global dependencies between tokens. Unlike BERT [9], which randomly masks some tokens in a sequence and the model predicts the masked tokens at random

positions, GPT always predicts the next masked token given preceding tokens, also known as auto-regressive training [14]. This guarantees that the prediction of EEG embeddings considers the causal temporal relationship between tokens, thus improving our model of the underlying brain activity patterns.

2.2. Pre-training

Pre-training Dataset: The large-scale public dataset, Temple University Hospital (TUH) EEG corpus, is used as the pre-training dataset. TUH EEG corpus comprises a diverse archive of clinical EEG recordings from 14,987 subjects with multiple sessions. The archive has over 40 different channel configurations and varying duration of recordings [11]. The sample frequency ranges from 250 - 1024 Hz, with the majority of recordings sampled at 250 Hz.

Preprocessing: We preprocessed the TUH EEG dataset using the Brainstorm software [15] in MATLAB (Mathworks, Inc.). Based on the channel labels, we selected 22 channels corresponding to the extended international 10-20 system (Fp1, Fp2, F7, F3, Fz, F4, F8, T1, T3, C3, Cz, C4, T4, T2, T5, P3, Pz, P4, T6, O1, Oz, O2). Channels with zero or missing signals throughout the recording sessions were marked as bad channels. The signals of the bad channels were interpolated by a weighted average of all neighboring channels with a maximal distance of 5cm between neighbors. EEG recordings were re-referenced to the average of 22 channels. We removed power line noise (60 Hz) using a notch filter and bandpass-filtered the data (0.5-100 Hz). All recordings were re-sampled to 250 Hz. We performed a DC offset correction and removed linear trends from the data. A z-transform was applied along the time dimension within each recording to normalize the data.

Implementation Details: During the pre-training phase, we simultaneously pre-train the entire Neuro-GPT model. After experimenting with various input configurations, we set the standard input as: 32 chunks, each with a length of 2 seconds and a 10% (0.2 second) overlap. We randomly select a starting point for each EEG recording and then sample 32 contiguous chunks. If the total length of the EEG recording is shorter than the length to be sampled (57.8 seconds), we apply zero-padding to the end of the sequence. The attention weights are set to zero for the zero-padded part. In each training batch, one sampled sequence is considered as a single training sample. The EEG encoder consists of two convolutional layers followed by six self-attention layers, with an embedding dimension of 1,080. The first convolutional layer has a kernel size of (1, 25), while the second has a kernel size of (C , 1), with C being the number of channels [13].

We employ the open-source GPT-2 [10] model provided by Hugging Face [16], which has an embedding dimension of 1024. We specify 6 transformer decoder layers in the GPT-2 model. A linear layer is added before the GPT-2 model to project the embedding dimension from 1080 to 1024. We pre-

processed 20,000 EEG recordings from the TUH EEG dataset with a total duration of 5656 hours. We train the model on 19,000 EEG recordings for 135 epochs. The remaining 1000 EEG recordings were used as a hold-out validation set.

2.3. Downstream Fine-tuning

Downstream Dataset: We define the downstream task as motor imagery classification, using the BCI Competition IV Dataset 2a provided by Graz University of Technology [17]. The BCI 2a dataset consists of nine subjects performing four motor imagery tasks: imagining left hand, right hand, feet, and tongue movement. Two sessions were collected on different days for each subject, using 22 Ag/AgCl electrodes at a sampling frequency of 250 Hz. Each recording has 72 trials per task, yielding a total of 288 trials. All trials from both sessions were used as training or testing samples - importantly, no subjects in the training data were included in the testing. Data was bandpass-filtered between 0.5 Hz and 100 Hz and normalized across time for each trial. We extract the sequence from $t = 2s$ to $t = 6s$ for each trial, which corresponds to the period when the cue and motor imagery tasks are performed.

Channel resampling: The downstream dataset has a different subset of 22 channel locations on the scalp from the pre-training dataset. To match the channel configuration between the two datasets, we resampled the downstream data to the pre-training dataset channel configuration using a 22×22 transformation matrix. The transformation matrix was computed by solving the forward and the inverse problem for the source localization, mapping from one sensor configuration to the cerebral cortex and then back to the second configuration [18, 19].

Fine-tuning Details: We fine-tune the pre-trained model on the BCI 2a dataset for the 4-class motor imagery classification task. To fully explore the potential of the foundation model, we designed three fine-tuning strategies:

1. **Encoder-only:** Remove the GPT model and fine-tune the pre-trained EEG encoder only. (Note that in this case the model still benefits from including GPT in pre-training through the self-supervised training of the encoder in combination with the GPT model.)
2. **Encoder+GPT:** Fine-tune the entire Neuro-GPT model.
3. **Linear:** Remove the GPT model, fix the EEG encoder and fine-tune only the linear head (3 linear layers).

All strategies use the same pre-trained model and involve adding a linear head consisting of 3 linear layers to the end of the model for classification. For the Encoder+GPT strategy, we maintain the same number of chunks, the same chunk length, and the same overlapping ratio as used in the pre-training stage. Since only a 4-seconds sequence is extracted from each EEG recording in the BCI 2a dataset, we apply zero-padding to the end of the sequence. In the Encoder-only strategy, we feed the model with two non-overlapping 2-second chunks, and no zero-padding is applied. For the

Linear strategy, all the pre-trained parameters from the EEG encoder are frozen during fine-tuning. We only fine-tune the linear head, which takes the output features of the EEG encoder as input. No masking is applied during fine-tuning.

3. EXPERIMENTS AND RESULTS

Fine-tuning Classification Performance: Unlike previous studies which only focused on within-subject classification [20, 13], we performed leave-one-subject-out cross-validation, which is more challenging due to the high inter-subject variance. We compute the average classification accuracy across subjects. To explore the benefits of applying a pre-trained foundation model, we compare the classification performance of a model trained from scratch (w/o pre-training) to that of the same model fine-tuned on the pre-trained foundation model (w/ pre-training). In addition, we compare the proposed Neuro-GPT with BENDR [8], a BERT-inspired transformer model trained on TUH EEG data using contrastive self-supervised learning and then fine-tuned on the BCI classification data. As shown in Table 1, Neuro-GPT significantly improved the classification performance compared with the best performance of BENDR, and outperforms other methods for motor imagery classification using leave-one-subject-out cross-validation.

The performance of models with pre-training surpassed that of models without pre-training for both Linear and Encoder-only fine-tuning strategies, highlighting that applying a foundation model to a downstream task can lead to effective feature learning and, consequently, improved performance. Among the fine-tuning strategies, Encoder-only achieved the best performance, indicating that the encoder learned expressive and generalizable features during pre-training, thus facilitating the learning of distinguishable features for downstream tasks. The Encoder+GPT yielded worse performance, possibly because the GPT model only serves as

an auxiliary component to assist the EEG encoder in encoding meaningful features from raw EEG data. The GPT model has more trainable parameters than the encoder. Fine-tuning a large model on a small data-set can lead to over-fitting. To examine whether the features learned by the foundation model are linearly separable, we input the features generated by the EEG encoder to the linear head for classification. The classification accuracy achieved by fine-tuning only the linear head is 0.443 vs. 0.398 with out pre-training, indicating that the EEG encoder can encode meaningful features through pre-training.

Hyper-parameter Evaluation in Pre-training: To explore the optimal input configurations for the foundation model during pre-training, we conducted experiments with varying numbers of chunks (4, 8, 16, 32), chunk lengths (1s, 2s, 4s), and overlapping ratios (10%, 50%). Different model architectures were also investigated. Key findings include:

- Chunks with a 1-second length are more straightforward to predict (as embedded tokens) but led to poorer downstream performance.
- Chunks with longer lengths are more challenging to predict but enhance downstream performance.
- Increasing the number of chunks is beneficial. Training with 32 and 16 chunks yielded better downstream results than training with 8 or 4 chunks.
- Increasing the overlapping ratio to 50% improved reconstruction, but degraded the downstream performance.
- Increasing the embedding dimension of GPT-2 model ($768 \rightarrow 1024$) improved downstream performance.
- Reducing the number of self-attention layers in the encoder ($6 \rightarrow 4, 2$) degraded downstream performance.
- Adding more GPT decoder layers ($6 \rightarrow 8, 10$) did not improve downstream performance.

4. DISCUSSION

We have demonstrated that pre-training a foundation model on a large-scale EEG dataset boosts downstream task performance. Through exploring different fine-tuning strategies, we discovered that the pre-trained EEG encoder captures inherent and fundamental features of EEG that are generalizable across datasets, leading to significant improvements in classification performance.

5. ACKNOWLEDGMENT

This project is sponsored in part by the NIH under grant R01 EB026299 and in part by the Defense Advanced Research Projects Agency (DARPA) under cooperative agreement No. N660012324006. The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

Table 1. A comparison of means and stds of four-class classification accuracy among different methods. The first three rows are three fine-tuning strategies of Neuro-GPT, accuracies reported in other work are shown in the bottom rows.

Method	w/o Pre-train	w/ Pre-train
Linear	0.398 ± 0.054	0.443 ± 0.051
Encoder-only	0.606 ± 0.098	0.645 ± 0.104
Encoder+GPT	0.596 ± 0.090	0.586 ± 0.098
BENDR [8]	/	0.426
SVM [21]	0.361 ± 0.082	/
EEGNet [22]	0.513 ± 0.052	/
CTCNN [23]	0.477 ± 0.151	/
CCNN [24]	0.553 ± 0.101	/
NG-CRAM [25]	0.601 ± 0.102	/

6. REFERENCES

- [1] Y. Du et al., “Eeg temporal–spatial transformer for person identification,” *Scientific Reports*, vol. 12, no. 1, pp. 14378, 2022.
- [2] C. J. Reed et al., “Self-supervised pretraining improves self-supervised pretraining,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 2584–2594.
- [3] L. Yu et al., “Language model beats diffusion – tokenizer is key to visual generation,” 2023.
- [4] K. Singhal et al., “Towards expert-level medical question answering with large language models,” 2023.
- [5] A. Thomas et al., “Self-supervised learning of brain dynamics from broad neuroimaging data,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 21255–21269, 2022.
- [6] J. Ortega Caro et al., “Brainlm: A foundation model for brain activity recordings,” *bioRxiv*, pp. 2023–09, 2023.
- [7] M. Azabou et al., “A unified, scalable framework for neural population decoding,” 2023.
- [8] D. Kostas, S. Aroca-Ouellette, and F. Rudzicz, “Bendr: using transformers and a contrastive self-supervised learning task to learn from massive amounts of eeg data,” 2021.
- [9] J. Devlin et al., “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.
- [10] A. Radford et al., “Language models are unsupervised multitask learners,” *OpenAI blog*, vol. 1, no. 8, pp. 9, 2019.
- [11] I. Obeid and J. Picone, “The temple university hospital eeg data corpus,” *Frontiers in neuroscience*, vol. 10, pp. 196, 2016.
- [12] C. Q. Lai et al., “Artifacts and noise removal for electroencephalogram (eeg): A literature review,” in *2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*. IEEE, 2018, pp. 326–332.
- [13] Y. Song et al., “EEG Conformer: Convolutional Transformer for EEG Decoding and Visualization,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 31, pp. 710–719, 2023.
- [14] T. B. Brown et al., “Language models are few-shot learners,” 2020.
- [15] F. Tadel et al., “Brainstorm: a user-friendly application for meg/eeg analysis,” *Computational intelligence and neuroscience*, vol. 2011, pp. 1–13, 2011.
- [16] T. Wolf et al., “Huggingface’s transformers: State-of-the-art natural language processing,” 2020.
- [17] C. Brunner et al., “Bci competition 2008–graz data set a,” *Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology*, vol. 16, pp. 1–6, 2008.
- [18] J. C. Mosher et al., “Eeg and meg: forward solutions for inverse methods,” *IEEE Transactions on biomedical engineering*, vol. 46, no. 3, pp. 245–259, 1999.
- [19] S. Baillet et al., “Electromagnetic brain mapping,” *IEEE Signal processing magazine*, vol. 18, no. 6, pp. 14–30, 2001.
- [20] C. Zhang et al., “Eeg-inception: an accurate and robust end-to-end neural network for eeg-based motor imagery classification,” *Journal of Neural Engineering*, vol. 18, no. 4, pp. 046014, 2021.
- [21] V. P. Oikonomou et al., “A comparison study on eeg signal processing techniques using motor imagery eeg data,” in *2017 IEEE 30th international symposium on computer-based medical systems (CBMS)*. IEEE, 2017, pp. 781–786.
- [22] V. J. Lawhern et al., “Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces,” *Journal of neural engineering*, vol. 15, no. 5, pp. 056013, 2018.
- [23] R. T. Schirrmeister et al., “Deep learning with convolutional neural networks for eeg decoding and visualization,” *Human brain mapping*, vol. 38, no. 11, pp. 5391–5420, 2017.
- [24] S. U. Amin et al., “Deep learning for eeg motor imagery classification based on multi-layer cnns feature fusion,” *Future Generation computer systems*, vol. 101, pp. 542–554, 2019.
- [25] D. Zhang et al., “Motor imagery classification via temporal attention cues of graph embedded eeg signals,” *IEEE journal of biomedical and health informatics*, vol. 24, no. 9, pp. 2570–2579, 2020.

Large GPT-like Models are Bad Babies: A Closer Look at the Relationship between Linguistic Competence and Psycholinguistic Measures

Julius Steuer Marius Mosbach Dietrich Klakow

Department of Language Science and Technology
Saarland University

{jsteuer, mmosbach, dietrich.klakow}@lsv.uni-saarland.de

Abstract

Research on the cognitive plausibility of language models (LMs) has so far mostly concentrated on modelling psycholinguistic response variables such as reading times, gaze durations and N400/P600 EEG signals, while mostly leaving out the dimension of what Mahowald et al. (2023) described as formal and functional linguistic competence, and developmental plausibility. We address this gap by training a series of GPT-like language models of different sizes on the strict version of the BabyLM pretraining corpus, evaluating on the challenge tasks (BLiMP, GLUE, MSGS) and an additional reading time prediction task. We find a positive correlation between LM size and performance on all three challenge tasks, with different preferences for model width and depth in each of the tasks. In contrast, a negative correlation was found between LM size and reading time fit of linear mixed-effects models using LM surprisal as a predictor, with the second-smallest LM achieving the largest log-likelihood reduction over a baseline model without surprisal. This suggests that modelling processing effort *and* linguistic competence may require an approach different from training GPT-like LMs on a developmentally plausible corpus.

1 Introduction

In recent years several approaches have been taken to test LMs for cognitive plausibility. This is usually done by using output probabilities of the LM as a predictor for a model’s preference towards certain linguistic structures (Roark et al., 2009; Wilcox et al., 2020). Another strain of research uses the output probabilities as a correlate of psycholinguistic measures, e.g., N400 and P600 EEG signals (Heilbron et al., 2019 and recently Li and Futrell, 2023) and (self-paced) reading times (Fernandez Monsalve et al., 2012). A natural question that arises is whether cognitive plausibility should be attributed to the model architecture itself, or to the training regime in combination with the training

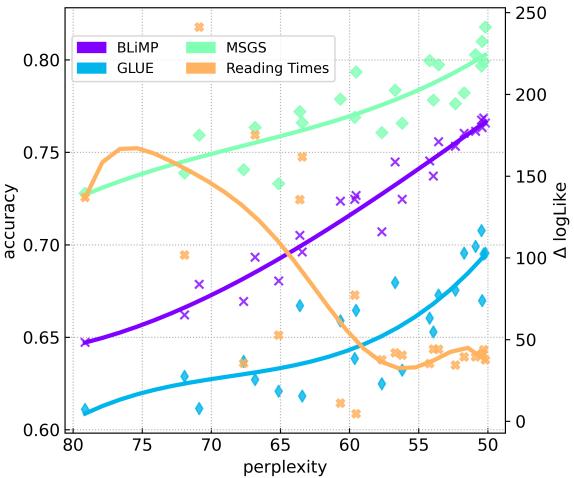


Figure 1: Our results show that LM performance on the BabyLM challenge tasks is negatively correlated with perplexity on the development set of the BabyLM corpus (lower perplexity leads to higher performance). In contrast, a *positive* correlation (Spearman’s $\rho = 0.4784$, $p < 0.05$) was found between LM perplexity and the fit of LM surprisal to self-paced reading times from the Natural Stories corpus (Futrell et al., 2021) in terms of the difference in log-likelihood between a baseline linear mixed-effects model and a model using LM surprisal as a predictor. Lines were fitted with 3 (challenge tasks) or 6 (reading times) degrees of freedom to the LMs’ average performance on the task. See Section 6 for detailed results.

dataset. Little research has been done on the actual neurological plausibility of large LMs (LLMs), but Schrimpf et al. (2021) showed that the architecture of BERT-like models is already plausible for the next word prediction task before training: model predictions with only the language modelling head trained are already predictive of human brain activity during reading *and* correlate well with the predictions of the fully trained model. In contrast, no correlation between brain activity and model predictions was found for models trained on GLUE (Wang et al., 2019), a natural language understand-

ing (NLU) benchmark. This finding may mirror an underlying difference in language processing between *formal* and *functional linguistic competence* as introduced by Mahowald et al. (2023):

Formal linguistic competence is defined as the "capacity required to produce and comprehend a given language, i.e., the ability to distinguish grammatically correct from incorrect formations, based either on "knowledge of and flexible use of linguistic rules" or "non-rule-like statistical regularities" (Mahowald et al., 2023). An example for the former mechanism would be the regular formation of past tense verbs in English (*look:looked*), and for the latter the formation of irregular or ablauting past tense verbs (*go:went,tread:trod*).

Functional linguistic competence is defined as "non-language-specific cognitive functions that are required when we use language in real-world circumstances" (Mahowald et al., 2023), i.e., the ability to perform cognitive tasks *with* language. GLUE is an example for a benchmark that test this dimension of linguistic competence, with some of its tasks (CoLA (Warstadt et al., 2019)) also testing for aspects of *formal linguistic competence*.

The dichotomy between formal and functional linguistic competence can be understood in terms of Wittgenstein's definition of the meaning of a word as its use in a language (Wittgenstein (1953), §43). The debate on whether statistical learners (i.e. LMs) can learn the meaning of a linguistic unit (word, phrase, text, etc.) in Wittgenstein's sense is still ongoing, with much division between positions that strongly deny that LMs can have such a property (Bender and Koller, 2020) and positions that advocate that they might have it, e.g., under the condition that the LM's predictions are grounded in extralinguistic reality (Bisk et al., 2020). Our study does not attempt to find arguments in favour of either position, but to study the implications of this dichotomy for the paradigm of cognitive modelling.

As stated earlier, the output probabilities of LMs lie at the basis of the application of LMs to cognitive language modelling, usually in the form of a probability distribution over a vocabulary of word forms given either surrounding words (masked language modelling) or preceding words (causal language modelling). Evidence for the use of surprisal (a word's negative logarithmic probability in con-

text) instead of the actual probability comes from logarithmic effects of contextual probabilities on processing difficulty (Shain et al., 2022). Another approach is to evaluate the output probabilities of a LM over a number of classes that may or may not apply to the input sequence, usually after fine-tuning the LM. The reliance of research in this direction on the output probabilities of LMs has already been criticized from multiple sides. There is a growing body of evidence that the performance of a LM in the typical language modelling task, next word prediction, and measures of formal linguistic competence are not correlated. Hu et al. (2020) found no correlation between LM perplexity and measures of formal linguistic competence, while Huang et al. (2023) argue that LM surprisal should not be assumed to be a good predictor of psycholinguistic measures of processing difficulty that require more than just lexical information. This lack of correlation with psycholinguistic measures becomes more prominent with the increasing size of LMs (Oh and Schuler, 2022), and especially so in extreme cases of human processing difficulty: Arehalli et al. (2022) showed that surprisal from LSTM-based LMs underestimates garden-path effects on reading times, while successfully predicting reading times for most non-garden-path sentences. This finding has been corroborated for transformer-based LMs such as GPT-2 (Jurayj et al., 2022) and BERT (Irwin et al., 2023).

2 BabyLM

The BabyLM challenge (Warstadt et al., 2023) introduces a novel constraint to cognitively plausible language modelling by limiting the token budget for LM pretraining to 100 million (100M) tokens, roughly the same amount of tokens a 13-year old child has seen during language acquisition (Gilkerson et al., 2017). While the focus of the challenge is on the pretraining procedure, the evaluation pipeline consists of the BLiMP (Warstadt et al., 2020a), MSGS (Warstadt et al., 2020b) and GLUE benchmarks, each of which aims to test for a specific dimension of linguistic competence.

BLiMP BLiMP tests for *formal linguistic competence* by comparing model predictions at a critical word in pairs of grammatically acceptable and unacceptable sentences, with the sentence pair only differing with respect to a single feature, e.g., whether a determiner agrees with its antecedent in gender or not. A model succeeds at the task if it assigns a

higher probability to the critical word in the acceptable sentence.

GLUE GLUE is a benchmark that requires fine-tuning¹ of the LM. It tests for a wide range of NLU problems, e.g., question answering, natural language inference and linguistic acceptability judgments, and hence can be regarded as a proxy for the *functional linguistic competence* of a LM.

MSGs MSGS is a benchmark of binary classification tasks that tests whether a LM prefers *surface generalizations* over *syntactic generalization* by first fine-tuning on data consistent with both types of generalization. At inference time, items are consistent with only one type, potentially revealing a bias towards either generalization type.

Previous studies mainly provided insights into the relationship of pretraining token budget and measures of formal and functional linguistic competence. Zhang et al. (2021) showed that encoder-only LMs already perform well on formal tasks such as BLiMP at a budget of 10-100M tokens, while requiring substantially larger token budgets to perform well on functional tasks such as GLUE. While this research established correlations for pretraining token budgets, similar relationships for *model size* at a fixed token budget have not yet been investigated. This study is dedicated to finding a relationship between model size and performance on these tasks, while simultaneously addressing the dimension of *processing effort*, which is not covered by the challenge tasks. This is done using the **strict** version of the BabyLM corpus, mainly because there is evidence that the fit with psycholinguistic measures profits from token budgets far larger than the 100M tokens in the corpus (Oh and Schuler, 2023). However, we also implicitly evaluate on models that are trained on token budgets of 10M tokens, corresponding rather to the **strict-small** track in Section 7.

3 Research questions

The starting point of our work is Zhang et al. (2021)'s finding of an earlier saturation effect (in terms of pretraining tokens) for BLiMP as opposed to (Super)GLUE. If performance on BLiMP is already close to the optimum after pretraining for

¹During fine-tuning, we train all parameters of the pre-trained LM as well as a randomly initialized classifier on top of the LM.

100M tokens, we suspect that a model with relatively small capacity is sufficient to reliably learn the required syntactic and semantic features. In contrast, the larger pretraining token budget and model size needed for GLUE should also require a model with higher capacity.

Studies on reading time prediction generally use causal LMs trained on a next-word prediction task instead of masked LMs (Oh and Schuler, 2022; Arehalli et al., 2022; Jurayj et al., 2022) because of their closer similarity to human language processing. Although masked LMs such as BERT show some word order effects (Papadimitriou et al., 2022) and even garden-path effects (Irwin et al., 2023), they are cognitively implausible in the sense that they process all words in a sequence simultaneously when predicting a word at a masked position, rather than processing language sequentially. This *autoregressive* property mirrors human language processing, and is therefore desirable in studies with the primary goal of modelling human reading behaviour. We therefore employ decoder-only, GPT-like LMs (Radford et al., 2019) in our study, i.e., we want to answer the following research questions:

Research question A

Are GPT-like models cognitively plausible in the sense that they are able to acquire (a degree of) formal and functional linguistic competence, while being also predictive of human processing effort?

Research question B

Can such LMs be trained on the same data as a child has available during language acquisition (100M tokens)?

4 Previous work

Do we need transformers for cognitive plausibility? Despite promising findings by Hosseini et al. (2021), it has yet to be determined whether transformers, and decoder-only transformer LMs in particular, are cognitively plausible in the sense that they are data-efficient enough to acquire human-like² linguistic competence. Indeed, there are results that seem to partially contradict the necessity

²Here, we do not use "human-like" to imply human-level performance, but rather that the model is *subject to similar processing constraints* as a human.

of LLMs with wide context windows in order for a model to exhibit human-like processing behaviour. [Kuribayashi et al. \(2022\)](#) showed that *reducing* context length of LLMs improves the fit of a linear mixed-effects model (LME) on gaze durations, with surprisal from a bigram GPT-2 model as a predictor yielding the largest log-likelihood reduction over the baseline model. [Wilcox et al. \(2020\)](#) failed to identify a relationship between psychometric predictive power (Δ log-likelihood) and syntactic generalization, concluding that different models are needed for modelling human processing effort versus syntactic generalization.

Linguistic competence vs. psycholinguistic measures It has long been clear that LM capacity, and subsequently LM perplexity, does not necessarily correlate with human-likeness ([Kuribayashi et al., 2021](#)). LLMs such as GPT-3 in particular were found to have considerable disadvantages when it comes to predicting psycholinguistic measures from their next-word predictions: [Oh and Schuler \(2022\)](#) found an inverse relationship between both perplexity and LLM capacity, versus fit to human reading times. The authors of this study hypothesize that this is because transformers have access to the full sequence context, and are trained on large enough corpora to make use of the information that they contain. This relationship between model perplexity and reading times is however not intrinsic to transformer-based LMs: [Hu et al. \(2020\)](#) found a similar relationship for LSTM LMs, though small GPT-like models have an advantage over recurrent models.

The impact of LM size on linguistic competence was investigated by [Eldan and Li \(2023\)](#), who found that relatively small GPT2-like models (<10M parameters) manage to produce fluent English and can be trained on relatively small corpora with a reduced vocabulary. Their study also shows that the relationship still holds for small models, while also identifying trade-offs between model width (hidden size) and depth (number of decoder layers).

As for training dataset size, [Oh and Schuler \(2023\)](#) found that surprisal from transformer-based LLMs gives the best fit to reading times at about 2B train tokens, across a wide range of model sizes. The corpus used in their study is very large (300B tokens), allowing for extensive training of a model without repeating any data. Reaching the same number of update steps with the much smaller

BabyLM corpus would require training for multiple epochs.

Single- vs. multi-epoch training Since the BabyLM training data is substantially smaller than the 2B tokens suggested by [Oh and Schuler \(2023\)](#), training our models in a multi-epoch setting cannot be avoided. Previous research has shown that repeating the training data can have adverse effects: [Xue et al. \(2023\)](#) compared single-epoch vs. multi-epoch training in a limited data setting and show that multi-epoch training leads to overfitting, with little performance being gained after the first epoch. They also find that regularization can only partially alleviate the overfitting problem, with dropout having the largest effect. Not having to repeat the training data is advantageous for downstream tasks and psycholinguistic modelling, if a certain amount of training data is available: [Oh and Schuler \(2023\)](#) found that reading time fit deteriorates after 2B tokens over a wide range of model sizes. However, it is not clear if repeating the training data would lead to an even stronger deterioration. If the corpus is substantially smaller than 2B tokens, repeating the training data could have a different effect, especially if the optimum of the reading time fit depends on the availability of the 2B tokens.

5 Methodology

Modelling We use the OPT architecture by [Zhang et al. \(2022\)](#) with a language modelling head for pretraining. Following our intuition that BLiMP should require much smaller model sizes than MSGS and GLUE, we train a series of OPT models of different sizes, varying only model width (hidden size) and model depth (number of decoder layers). In total we train 24 models varying over 4 hidden sizes $l_{\text{hidden}} \in \{192, 384, 768, 1536\}$ and 6 numbers of decoder layers ($l_{\text{decoder}} \in \{1, 2, 4, 8, 16, 24\}$). We also adjust the dimension of the feedforward layers such that the size of the output vector $l_{\text{forward}} = 3 \times l_{\text{hidden}}$. Table 1 in Appendix A shows the resulting model sizes. The models and all code for pretraining are implemented with PyTorch ([Paszke et al., 2019](#)) and HuggingFace transformers ([Wolf et al., 2020](#)), starting from their implementation of OPT. We also trained a new tokenizer on the training set of the BabyLM corpus, using the same vocabulary size $|V| = 50272$ as the original OPT tokenizer. We report all results as averages over 3 random seeds (see Appendix D for full results and standard error).

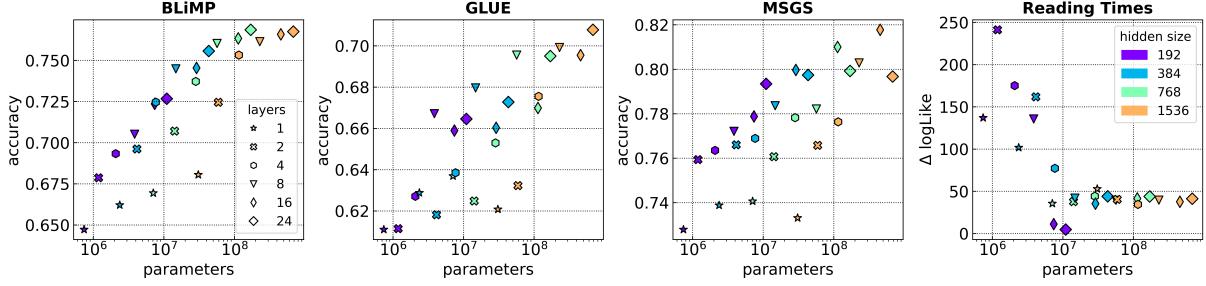


Figure 2: Task performance by model size (higher numbers are better). Baselines can be found in Appendix D.

Training Following the Shortformer pipeline (Press et al., 2021), each model is trained for one epoch with an initial sequence length of 64, followed by 4 epochs with the full sequence length of 256. The full sequence length of 256 was chosen as a compromise between the relatively short test items in the challenge tasks (up to 128 tokens). In order to ensure that the model generalizes to longer sequences we use ALIBI (Press et al., 2022) instead of learned positional embeddings. This also ensures that our models generalize to the longer sequences in the Natural Stories corpus. We trained each model on a A100 GPU with 40 GB VRAM and an effective batch size of 128, using gradient accumulation for models that could not fit the full batch size. We used AdamW (Loshchilov and Hutter, 2019) as our optimizer with an initial learning rate of 0.001 and weight decay of 0.001 with 2000 linear warm-up steps. We use a dropout of 0.1 following the default HuggingFace transformers parameters for OPT.

Pretraining experiments We also experimented with changes to the pretraining regime. We trained models on multiple permutations of the training dataset: ordering sequences according to length (number of words), word length (number of characters), sequence-level perplexity from a 3-gram LM trained on the same data, and different orderings of the subcorpora as in Mueller and Linzen (2023). None of these approaches resulted in significant performance gains in terms of perplexity and performance on the challenge tasks over a baseline model trained on the concatenated BabyLM corpus with subsequent shuffling of the sequences.

Evaluation We evaluated all models on the downstream tasks of the BabyLM challenge. While these three tasks test for the linguistic competence of a model, they do not quantify the cognitive effort associated with language

processing. We therefore also evaluate all models on a reading time prediction task. For each model, we calculated surprisal on the items of the Natural Stories Corpus (Futrell et al., 2021). This corpus was chosen because its domain is close to at least one of the BabyLM subcorpora (Children’s Stories). We fitted linear mixed-effects (LME) models with random intercepts for subject, word and item (the id of the story); surprisal, word frequency, word length and sentence position as predictors and log-normalized reading times as the response variable. The exact formula is

$$\begin{aligned} \log(\text{reading_time}) \sim \\ & \text{word_surprisal} + \text{len(word)} \\ & + \log(\text{word_frequency}) + \text{position} \\ & + (1/\text{word}) + (1/\text{subject}) + (1/\text{item}) \end{aligned}$$

For the reading time analysis we report the difference in log-likelihood between the models with surprisal as a predictor over a baseline model with only the control predictors. For all other tasks we report accuracy.

Code We used the evaluation code provided by the organizers of the BabyLM challenge³, with some modifications to load custom models. The evaluation pipeline is based on the LM-Eval framework by Gao et al. (2021). Fine-tuning on GLUE and MSGS was done with the default hyperparameter settings, but we reduced the number of fine-tuning epochs to 3 as we did not observe any improvements after 3 epochs. The LME models were fitted using the lmerTest R library (Kuznetsova et al., 2017) via the pymer4 Python package (Jolly, 2018). The code to pretrain and evaluate all models is publicly available on GitHub⁴. The model with the highest BLiMP accuracy and detailed results for the LME models are made available at the same

³<https://github.com/BabyLM/evaluation-pipeline>

⁴<https://github.com/uds-lsv/babylm>

location, alongside instructions on how to run the training and evaluation pipelines.

6 Results

Fine-tuning GLUE Fine-tuning on GLUE was overall very unstable and often failed to outperform the baseline. This was mainly due to the one-size-fits-all approach to the fine-tuning hyperparameters; we repeated several more fine-tuning runs with different hyperparameter settings on some of the GLUE tasks, and found that, e.g., RTE profited from a longer warm-up period (which is in line with the findings of Mosbach et al. (2021) for BERT-like models), but most other sub-tasks fine-tuned with the same hyperparameters showed a drop in performance. While we could have optimized hyperparameters for all sub-tasks, the main objective of the BabyLM challenge is to improve the pretraining part of the NLP pipeline. Thus, we decided to fine-tune with the default hyperparameters, only adjusting the number of epochs as we found that the fine-tuning runs already converged after a few epochs.

Model size Figure 2 shows the relationship between model size and task performance: While GLUE (Spearman’s $\rho = 0.7739, p < 1^{-4}$) and MSGS ($\rho = 0.7148, p < 1^{-4}$) performance scales with model size, BLiMP performance plateaus after reaching a model size of about 50M parameters ($\rho = 0.8835, p < 1^{-4}$). In contrast, reading time fit was negatively correlated with model size ($\rho = -51.39, p < 0.05$). All correlations are statistically significant with $p < 1^{-4}$. No single model performed best on all three challenge tasks, with large differences in the size of the best model. Figure 1 shows that similar correlations hold for model perplexity and task performance (BLiMP: $\rho = -0.9765, p < 1^{-4}$, GLUE: $\rho = -0.8287, p < 1^{-4}$, MSGS: $\rho = -0.8661, p < 1^{-4}$); negative correlations mean that lower perplexity leads to higher performance. We found strong positive correlations (pictured in Figure 7 in Appendix D) between performance on the challenge tasks (BLiMP and GLUE ($\rho = 0.8784$), BLiMP and MSGS ($\rho = 0.9182$) and GLUE and MSGS ($\rho = 0.815$) generally with $p < 1^{-4}$).

Model width vs. depth While BLiMP performance was not found to be strongly correlated with either the number of decoder layers or hidden size, GLUE and MSGS showed some variability based

on the number of layers. For GLUE the only configuration that showed a monotonic improvement in performance was a hidden size of 1536, with models with more decoder layers achieving higher accuracy in this setting. For MSGS we observed a drop in performance for the models with 24 decoder layers at the largest hidden sizes (384, 768). Overall, the effect of hidden size and number of layers was minor when compared to overall model size. In contrast, the best fit on the reading time data was achieved with the second smallest model with only 2 decoder layers and a hidden size of 192. Figure 3 illustrates this trend: for the challenge tasks, performance increases with the number of layers (though not monotonically), whereas Δ log-likelihood of the LME models decreases with the number of layers at $l_{hidden} = 192$ and, to a lesser extent, at $l_{hidden} = 384$, while deeper models with more decoder layers and larger hidden sizes perform considerably worse.

Possible confounds The reading time analysis suffers from several potential confounding factors: Firstly, the domain of the training data differs considerably from the data in the Natural Stories corpus. While the training data also contains some longer texts (Wikipedia, Children’s Stories), most of the corpora are more representative of spoken language (Open Subtitles, BNC Spoken, CHILDES). In addition, most sequences are relatively short, with a median sequence length of 8 in the Open Subtitles corpus, which accounts for >50% of the training data. This is considerably less than the median sequence length of 22 in the Natural Stories corpus. Another confounding factor might be the difference in exposure to language data of the model and that of the participants of the original reading time study. Futrell et al. (2021) do not provide demographic data of their participants, but since data collection was done via Amazon Mechanical Turk we can safely assume that the mean age of the participants was higher than 13, meaning that they were exposed to considerably more language data than the 100M tokens in the BabyLM corpus. Although a recent study by Oh and Schuler (2023) showed that reading time fit (in terms of Δ log-likelihood) from transformer models still profits from pretraining data multiple orders of magnitude larger than our corpus, with an optimum at 2B tokens, this is partially alleviated in this study by the multiple-epoch training regime, totalling about 500M tokens seen by each of our

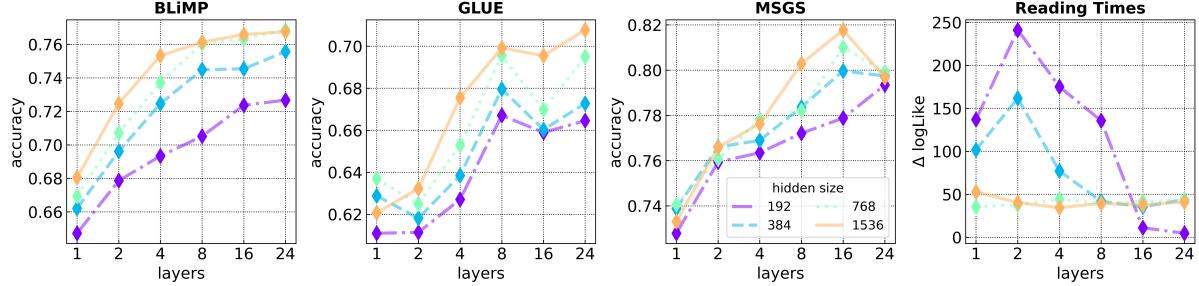


Figure 3: Task performance by hidden size, number of layers and task.

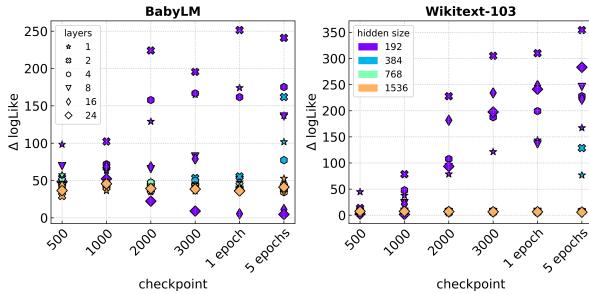


Figure 4: Reading time fit in terms of $\Delta \log\text{-likelihood}$ over a base model without surprisal as a predictor, on the BabyLM and Wikitext-103 data after 500, 1000, 2000 and 3000 training steps (1/8, 1/4, 1/2 and 3/4 of an epoch) and 1 and 5 epochs.

models. Since Oh and Schuler (2023) found that training on more *unseen* tokens after reaching the optimum leads to a quick deterioration of reading time fit proportional to model size, it is unclear what impact repeating the training data would have on the reading time fit.

7 Reading time prediction in a multi-epoch setting

Experimental setup In order to evaluate whether the negative correlation is an artifact of the domain mismatch between the BabyLM corpus and the items in the Natural Stories corpus or the repetition of the training data before reaching the optimal token budget, we conduct two additional experiments: First, we retrain all models on the BabyLM corpus for a single epoch, saving intermediate checkpoints at 100, 500, 1000, 2000 and 3000 training steps. Then, we use the intermediate models to fit LME models to the reading time data, using the same formula as given in Section 5. Second, we replicate these experiments on Wikitext-103, a corpus of similar size that does not have the same limitations of the BabyLM corpus (i.e. an average sequence length and a domain closer to the Natural Stories

corpus). The models trained on Wikitext-103 serve as a control for the experiments on the BabyLM corpus and were not included in the final submission. Since the results indicate that larger models yield a worse reading time fit, we restrict the experiment to small models (1-4 layers, all hidden sizes) and larger models with the smallest and largest hidden size (192 and 1536). The models are trained with the same hyperparameter settings as the original models, but sequence length is not reduced in the first epoch.

Results Figure 4 shows a somewhat different picture for the models trained on Wikitext-103, with reading time fit of smaller models increasing over the whole pretraining procedure, while models with $l_{\text{hidden}} > 192$ almost never improve over the baseline model. In contrast, the reading time fit of the LMs trained on the BabyLM data improves significantly over the baseline for shallower models (< 2 decoder layers), while staying roughly constant for deeper and wider models (16, 24 decoder layers). However, the relationship between the number of training steps and reading time fit is not monotonic, with a slight decrease after training for 4 more epochs for the best model. While the models trained on the Wikitext-103 dataset yield a better fit to reading times in terms of $\Delta \log\text{-likelihood}$, the basic finding on the BabyLM data is corroborated: exposing a transformer model to multiple repetitions of the training data before reaching the optimal token budget does not lead to a decrease in reading time fit, but also does not improve over the single epoch setting in a meaningful way. The results also show that the improved reading time fit for $l_{\text{hidden}} = 192$ cannot be attributed to smaller model size alone, as the deepest model with that hidden size, 24*192 shows an improved fit over the baseline, while 1*384, a model with a comparable number of parameters, but a larger hidden size,

does not. In conclusion, we did not find a degradation of reading time fit when repeating the training data, with similar effects of LM size on reading time fit for Wikitext-103 and the BabyLM corpus (see Table 2 in Appendix C for Spearman’s ρ ’s and p-values). We also found The BabyLM corpus to be advantageous for this task in the sense that – in contrast to Wikitext-103 – reading time fit from all models improved over the baseline LME model.

8 Discussion

Correlation between BLiMP, GLUE & MSGS

The experiments presented in Section 6 provide evidence for a correlation between LM performance on BLiMP, GLUE and MSGS tasks when pretraining on the BabyLM corpus. This correlation is in accordance with established effects of training dataset size (Zhang et al., 2021), and interactions of train corpus size and model capacity (Eldan and Li, 2023, Kaplan et al., 2020). However, no single model achieves the highest score on all three tasks: BLiMP shows diminishing returns for model sizes larger than 50M tokens, while the best model on MSGS (16*1536) is substantially smaller than the best model on GLUE (24*1536). This discrepancy between the best model on the BabyLM challenge tasks and on the reading times prediction task is illustrated by Figure 5. The correlation between BLiMP/MSGS and GLUE may be an artifact of the sub-optimal fine-tuning on GLUE, failing to outperform the baseline model. It cannot be ruled out that the results would change when determining the optimal hyperparameters for each sub-task individually. However, even if the correlation were an artifact of the pretraining data, the findings of a negative correlation between model size and reading time fit would still hold.

Cognitive plausibility of GPT-like models The best fit on self-paced reading times from the Natural Stories corpus was obtained with the second-smallest model, with models with $l_{hidden} > 192$ only slightly improving over the baseline. The second suite of experiments in Section 7 confirms that this is not solely caused by the multi-epoch training regime necessitated by the small token budget. The reason for the mismatch between measures of cognitive plausibility (reading times) and measures of formal (BLiMP, MSGS) and functional linguistic competence (GLUE) is rooted in the interaction of pretraining regime and model size: While it is feasible to train a model that performs com-

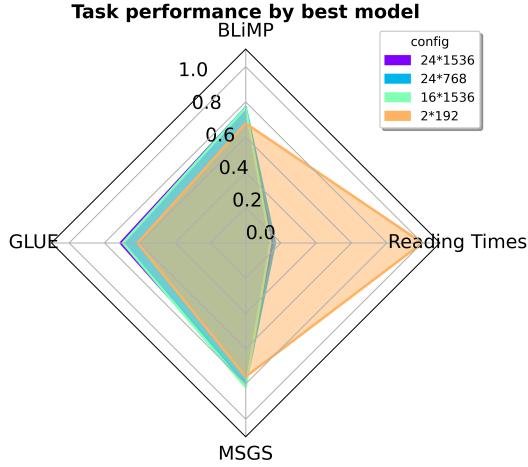


Figure 5: Performance of the best models by task. Reading times Δ log-likelihoods are normalized in the interval [0, 1].

paratively well on all four tasks on a budget of 100M tokens, the sweet spot for model size and dataset size is reached much earlier for the reading time prediction task than for the BabyLM challenge tasks. This problem could easily be resolved by using one model when modelling reading times (or any other psycholinguistic measure), and another model when either of the forms of linguistic competence is the aim. This might be a valid and promising approach in a situation where the understanding of the research object does not depend on the connectedness of its experimental analogs. In the case of our research object – the human language faculty – it may not be necessary to find a single analogon that accounts for all its components, but since we *know* that the human language faculty is part of a unified cognitive system (with specialized sub-units) performing the tasks which the modern language modelling pipeline of pre-training and fine-tuning splits up into individual modules, it would be worthwhile to move in the direction of a unified approach that accounts for both forms of linguistic competence and empirical evidence of processing effort. This could be achieved through adjustments to the pretraining regime (in terms of data, modelling objective etc.), as suggested by the BabyLM challenge, or through adjustments to the model architecture.

Size of transformer models The results of the reading time prediction study on the BabyLM corpus indicate that it in fact has an *advantage* over

Wikitext-103, although the LMs trained on the latter achieve larger Δ log-likelihoods on average: Since the largest models fail to improve over the baseline model if trained on Wikitext-103, it is possible that some properties of the language in the BabyLM corpus facilitate the learning mechanism that actuates the correlation of LM surprisal and reading times. The reason for the worse fit of surprisal from the larger models may be that both Wikitext-103 and the BabyLM corpus are not large enough to induce the learning bias needed to give good predictions of reading times in larger models, with Figure 4 showing that the results on the BabyLM corpus are much less stable than on Wikitext-103 and the improvements over the baseline much less sharply linear. In summary, our results lead to the following answers to our research questions:

Result: Research question A

GPT-like LMs can be cognitively plausible and display formal and functional linguistic competence, although not both at the same time...

Result: Research question B

...under the constraint of a developmentally plausible training dataset.

9 Conclusion

Our study highlights the challenges of training a LM that performs well on tasks requiring some degree of formal and functional linguistic competence as defined by [Mahowald et al. \(2023\)](#), while also being predictive of the psycholinguistic measure of reading times. We find that small, shallow models of less than 5M parameters yield the best fit to the psycholinguistic measure, while performance on BLiMP, GLUE and MSGS improves with increasing model size, although to a different degree for each of the tasks. This has implications for research on cognitively or developmentally plausible models of human language processing: in the case of a small, domain-specific training corpus it is not feasible to pretrain an LLM that displays formal linguistic competence and performs well on a reading time prediction tasks, a conclusion also drawn by [Wingfield and Connell \(2022\)](#). Consequently, research in this direction has concentrated on fine-

tuning pretrained LLMs on domain-specific data, e.g., [Škrjanec et al. \(2023\)](#). A promising approach to a unified architecture could be relegating special tasks (such as classifying a sequence as in GLUE) to adapters ([Houlsby et al., 2019](#)), sub-networks within a pretrained LM. This approach is common in multilingual language modelling ([Pfeiffer et al., 2022; Alabi et al., 2022](#)), where its success is partially attributed to its ability to separate general linguistic knowledge from language-specific information. A similar modelling decision may be necessary for cognitively plausible language models.

Acknowledgements

The authors thank Iza Škrjanec for helping with the training and interpretation of LME models, and Vagrant Gautam, Michael Hahn, Benedict Schneider, Iza Škrjanec and for their helpful comments. This research was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), Project ID 232722074 – SFB 1102.

Limitations

The results of the paper mainly hold for decoder-only transformer LMs. While these LMs are closer to human language processing in the sense that they process language incrementally, this has some disadvantages for reading time predictions, since humans do not attribute equal importance to each word, skipping some words in the process, and typically integrate words from the left- *and* right-hand context of a fixated word. While the first point can be addressed by explicitly modelling skipping behaviour ([Hahn and Keller, 2016](#)), the second could require a solution closer to masked language models.

A second limitation is the focus on self-paced reading time as the psycholinguistic response variable. Since the setup of self-paced reading studies, with the participants observing a single word at a time, distorts the natural reading process, the measure itself may be not that cognitively plausible. This could be addressed by repeating the experiments on corpora from eye-tracking studies such as the Dundee corpus ([Kennedy and Pynte, 2005](#)). There is evidence that much larger models than those tested in the current study still improve the fit to total reading times in less restricted experimental settings ([de Varda and Marelli, 2023](#)). The latter study also shows that the fit to psycholinguistic measures varies over languages and writing

systems.

Another option is modelling brain activity patterns directly by predicting N400 and P600 EEG signals, which have the additional advantage of providing a means of decomposing LM surprisal without the proxy of linguistic structure, as shown by Li and Futrell (2023).

Ethics Statement

The authors foresee no ethical concerns about the work presented in the paper.

References

- Jesujoba O. Alabi, David Ifeoluwa Adelani, Marius Mosbach, and Dietrich Klakow. 2022. [Adapting pre-trained language models to African languages via multilingual adaptive fine-tuning](#). In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4336–4349, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.
- Suhas Arehalli, Brian Dillon, and Tal Linzen. 2022. [Syntactic Surprisal From Neural Models Predicts, But Underestimates, Human Processing Difficulty From Syntactic Ambiguities](#). In *Proceedings of the 26th Conference on Computational Natural Language Learning (CoNLL)*, pages 301–313, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Emily M. Bender and Alexander Koller. 2020. [Climbing towards NLU: On meaning, form, and understanding in the age of data](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5185–5198, Online. Association for Computational Linguistics.
- Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph Turian. 2020. [Experience grounds language](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8718–8735, Online. Association for Computational Linguistics.
- Andrea de Varda and Marco Marelli. 2023. [Scaling in cognitive modelling: a multilingual approach to human reading times](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 139–149, Toronto, Canada. Association for Computational Linguistics.
- Ronen Eldan and Yuanzhi Li. 2023. [TinyStories: How Small Can Language Models Be and Still Speak Coherent English?](#)
- Irene Fernandez Monsalve, Stefan L. Frank, and Gabriella Vigliocco. 2012. [Lexical surprisal as a general predictor of reading time](#). In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 398–408, Avignon, France. Association for Computational Linguistics.
- Richard Futrell, Edward Gibson, Harry J. Tily, Idan Blank, Anastasia Vishnevetsky, Steven T. Piantadosi, and Evelina Fedorenko. 2021. [The Natural Stories corpus: A reading-time corpus of English texts containing rare syntactic constructions](#). *Language Resources and Evaluation*, 55(1):63–77.
- Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2021. [A framework for few-shot language model evaluation](#).
- Jill Gilkerson, Jeffrey A. Richards, Steven F. Warren, Judith K. Montgomery, Charles R. Greenwood, D. Kimbrough Oller, John H. L. Hansen, and Terrance D. Paul. 2017. [Mapping the Early Language Environment Using All-Day Recordings and Automated Analysis](#). *American Journal of Speech-Language Pathology*, 26(2):248–265.
- Michael Hahn and Frank Keller. 2016. [Modeling human reading with neural attention](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 85–95, Austin, Texas. Association for Computational Linguistics.
- Micha Heilbron, Benedikt Ehinger, Peter Hagoort, and Floris de Lange. 2019. [Tracking Naturalistic Linguistic Predictions with Deep Neural Language Models](#). In *2019 Conference on Cognitive Computational Neuroscience*, Berlin, Germany. Cognitive Computational Neuroscience.
- Kasra Hosseini, Kaspar Beelen, Giovanni Colavizza, and Mariona Coll Ardanuy. 2021. [Neural Language Models for Nineteenth-Century English](#). ArXiv:2105.11321 [cs].
- Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. [Parameter-efficient transfer learning for nlp](#).
- Jennifer Hu, Jon Gauthier, Peng Qian, Ethan Wilcox, and Roger P. Levy. 2020. [A Systematic Assessment of Syntactic Generalization in Neural Language Models](#). Publisher: arXiv Version Number: 2.
- Kuan-Jung Huang, Suhas Arehalli, Mari Kugemoto, Christian Muxica, Grusha Prasad, Brian Dillon, and Tal Linzen. 2023. [Surprisal does not explain syntactic disambiguation difficulty: evidence from a large-scale benchmark](#). preprint, PsyArXiv.

- Tovah Irwin, Kyra Wilson, and Alec Marantz. 2023. **BERT Shows Garden Path Effects**. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3220–3232, Dubrovnik, Croatia. Association for Computational Linguistics.
- Eshin Jolly. 2018. **Pymer4: Connecting R and Python for Linear Mixed Modeling**. *Journal of Open Source Software*, 3(31):862.
- William Juraj, William Rudman, and Carsten Eickhoff. 2022. **Garden-Path Traversal in GPT-2**. ArXiv:2205.12302 [cs].
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. **Scaling Laws for Neural Language Models**. ArXiv:2001.08361 [cs, stat].
- Alan Kennedy and Joël Pynte. 2005. **Parafoveal-on-foveal effects in normal reading**. *Vision Research*, 45(2):153–168.
- Tatsuki Kurabayashi, Yohei Oseki, Ana Brassard, and Kentaro Inui. 2022. **Context Limitations Make Neural Language Models More Human-Like**. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10421–10436, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Tatsuki Kurabayashi, Yohei Oseki, Takumi Ito, Ryo Yoshida, Masayuki Asahara, and Kentaro Inui. 2021. **Lower Perplexity is Not Always Human-Like**. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5203–5217, Online. Association for Computational Linguistics.
- Alexandra Kuznetsova, Per B. Brockhoff, and Rune H. B. Christensen. 2017. **lmerTest package: Tests in linear mixed effects models**. *Journal of Statistical Software*, 82(13):1–26.
- Jiaxuan Li and Richard Futrell. 2023. **A decomposition of surprisal tracks the N400 and P600 brain potentials**. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 45(45).
- Ilya Loshchilov and Frank Hutter. 2019. **Decoupled weight decay regularization**.
- Kyle Mahowald, Anna A. Ivanova, Idan A. Blank, Nancy Kanwisher, Joshua B. Tenenbaum, and Evelina Fedorenko. 2023. **Dissociating language and thought in large language models: a cognitive perspective**.
- Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2021. **On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines**.
- Aaron Mueller and Tal Linzen. 2023. **How to Plant Trees in Language Models: Data and Architectural Effects on the Emergence of Syntactic Inductive Biases**. Publisher: arXiv Version Number: 1.
- Byung-Doh Oh and William Schuler. 2022. **Why Does Surprisal From Larger Transformer-Based Language Models Provide a Poorer Fit to Human Reading Times?** Publisher: arXiv Version Number: 1.
- Byung-Doh Oh and William Schuler. 2023. **Transformer-Based LM Surprisal Predicts Human Reading Times Best with About Two Billion Training Tokens**. ArXiv:2304.11389 [cs].
- Isabel Papadimitriou, Richard Futrell, and Kyle Mahowald. 2022. **When classifying grammatical role, BERT doesn't care about word order... except when it matters**. ArXiv:2203.06204 [cs].
- Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. **Pytorch: An imperative style, high-performance deep learning library**. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.
- Jonas Pfeiffer, Naman Goyal, Xi Lin, Xian Li, James Cross, Sebastian Riedel, and Mikel Artetxe. 2022. **Lifting the Curse of Multilinguality by Pre-training Modular Transformers**. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3479–3495, Seattle, United States. Association for Computational Linguistics.
- Ofir Press, Noah A. Smith, and Mike Lewis. 2021. **Shortformer: Better Language Modeling using Shorter Inputs**. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5493–5505, Online. Association for Computational Linguistics.
- Ofir Press, Noah A. Smith, and Mike Lewis. 2022. **Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation**. ArXiv:2108.12409 [cs].
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. **Language Models are Unsupervised Multitask Learners**.
- Brian Roark, Asaf Bachrach, Carlos Cardenas, and Christophe Pallier. 2009. **Deriving lexical and syntactic expectation-based measures for psycholinguistic**

- modeling via incremental top-down parsing. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 324–333, Singapore. Association for Computational Linguistics.
- Martin Schrimpf, Idan Asher Blank, Greta Tuckute, Cárina Kauf, Eghbal A. Hosseini, Nancy Kanwisher, Joshua B. Tenenbaum, and Evelina Fedorenko. 2021. The neural architecture of language: Integrative modeling converges on predictive processing. *Proceedings of the National Academy of Sciences*, 118(45):e2105646118.
- Cory Shain, Clara Meister, Tiago Pimentel, Ryan Cotterell, and Roger Philip Levy. 2022. Large-Scale Evidence for Logarithmic Effects of Word Predictability on Reading Time. preprint, PsyArXiv.
- Iza Škrjanec, Frederik Y. Broy, and Vera Demberg. 2023. Expert-adapted language models improve the fit to reading times.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. ArXiv:1804.07461 [cs].
- Alex Warstadt, Aaron Mueller, Leshem Chohen, Ethan Gotlieb Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Adina Williams, Bhargavi Paranjape, Tal Linzen, and Ryan Cotterell. 2023. Findings of the 2023 BabyLM Challenge: Sample-efficient pretraining on developmentally plausible corpora. In *Proceedings of the 2023 BabyLM Challenge*. Association for Computational Linguistics (ACL).
- Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Monahaney, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020a. BLiMP: The Benchmark of Linguistic Minimal Pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.
- Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments.
- Alex Warstadt, Yian Zhang, Xiaocheng Li, Haokun Liu, and Samuel R. Bowman. 2020b. Learning Which Features Matter: RoBERTa Acquires a Preference for Linguistic Generalizations (Eventually). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 217–235, Online. Association for Computational Linguistics.
- Ethan Gotlieb Wilcox, Jon Gauthier, Jennifer Hu, Peng Qian, and Roger Levy. 2020. On the Predictive Power of Neural Language Models for Human Real-Time Comprehension Behavior. Publisher: arXiv Version Number: 1.
- Cai Wingfield and Louise Connell. 2022. Understanding the role of linguistic distributional knowledge in cognition. volume 37, pages 1220–1270. Routledge.
- Ludwig Wittgenstein. 1953. *Philosophische Untersuchungen*. Suhrkamp Verlag, Frankfurt am Main.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierrette Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Fuzhao Xue, Yao Fu, Wangchunshu Zhou, Zangwei Zheng, and Yang You. 2023. To Repeat or Not To Repeat: Insights from Scaling LLM under Token-Crisis.
- Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuhui Chen, Christopher DeWan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Miaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: Open Pre-trained Transformer Language Models. Publisher: arXiv Version Number: 4.
- Yian Zhang, Alex Warstadt, Xiaocheng Li, and Samuel R. Bowman. 2021. When do you need billions of words of pretraining data? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1112–1125, Online. Association for Computational Linguistics.

A OPT models

$l_{decoder}$	l_{hidden}	Parameters (non-embedding)
1	192	0.74
2	192	1.19
4	192	2.07
8	192	3.85
16	192	7.41
24	192	10.9
1	384	2.37
2	384	4.14
4	384	7.69
8	384	14.79
16	384	28.99
24	384	43.18
1	768	7.09
2	768	14.18
4	768	28.35
8	768	56.70
16	768	113.41
24	768	170.11
1	1536	30.69
2	1536	59.00
4	1536	115.69
8	1536	229.01
16	1536	455.67
24	1536	682.32

Table 1: OPT models sizes in million parameters by hidden size and number of decoder layers. The number of parameters does not include the embedding table, which is always of the size $l_{emb} \times |V| = 768 \times 50272 = 38.608.896$, as in OPT-128m.

B Validation perplexity

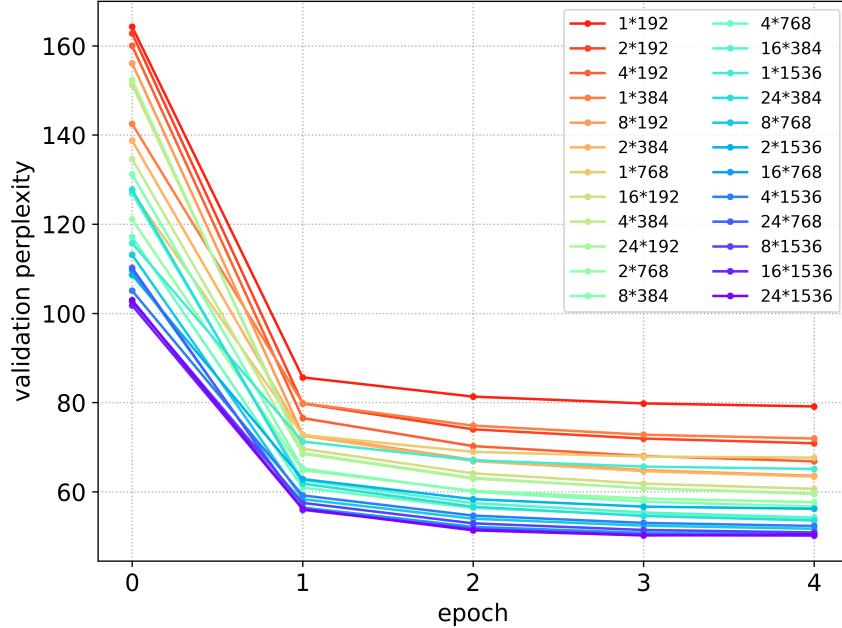


Figure 6: Validation perplexity by configuration and epoch on the development set of the BabyLM corpus.

C Detailed results: Reading time experiments

Corpus	Step	Spearman's ρ	p-value
babylm	500	-0.5913	0.0097
babylm	1000	-0.6285	0.0052
babylm	2000	-0.7833	0.0001
babylm	3000	-0.7874	0.0001
babylm	1	-0.7915	0.0001
babylm	5	-0.614	0.0067
wikitext-103	500	0.0815	0.7478
wikitext-103	1000	-0.4241	0.0794
wikitext-103	2000	-0.7482	0.0004
wikitext-103	3000	-0.7441	0.0004
wikitext-103	1	-0.7172	0.0008
wikitext-103	5	-0.7523	0.0003

Table 2: Spearman's ρ of model size (in terms of number of parameters) and Δ log-likelihood over the baseline LME model. Steps 1 and 5 refer to the first and fifth epoch.

D Detailed results: BabyLM challenge tasks

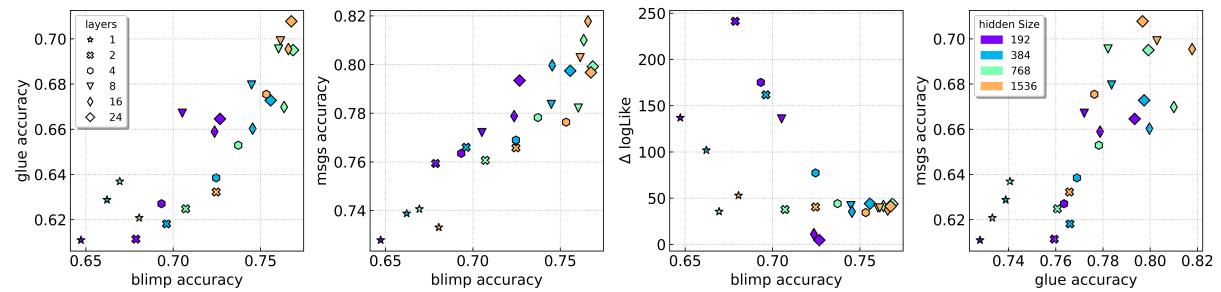


Figure 7: Correlation of LM performance on BLiMP vs. GLUE, BLiMP vs. MSGS, GLUE vs. MSGS.

Table 3: BLiMP accuracy by task and model

Task	OPT-LSMs (baseline)									
	1*10 ⁶	1*10 ⁶	2*10 ⁶	2*10 ⁶	2*10 ⁶	2*10 ⁶	4*10 ⁶	4*10 ⁶	8*10 ⁶	8*10 ⁶
anglo-s-argument	0.95 ± 0.01	0.96 ± 0.01	0.95 ± 0.00	0.94 ± 0.01	0.96 ± 0.00	0.97 ± 0.00	0.96 ± 0.00	0.97 ± 0.00	0.98 ± 0.00	0.99 ± 0.00
argument-structure	0.70 ± 0.00	0.71 ± 0.00	0.72 ± 0.00	0.73 ± 0.00	0.74 ± 0.00	0.75 ± 0.00	0.76 ± 0.00	0.77 ± 0.00	0.78 ± 0.00	0.79 ± 0.00
biasing	0.67 ± 0.01	0.68 ± 0.01	0.68 ± 0.00	0.69 ± 0.01	0.69 ± 0.02	0.72 ± 0.02	0.71 ± 0.02	0.72 ± 0.02	0.73 ± 0.02	0.74 ± 0.02
controlling	0.64 ± 0.01	0.66 ± 0.01	0.67 ± 0.00	0.67 ± 0.01	0.69 ± 0.00	0.72 ± 0.00	0.73 ± 0.00	0.73 ± 0.00	0.74 ± 0.00	0.74 ± 0.00
declaiming	0.85 ± 0.00	0.85 ± 0.00	0.88 ± 0.00	0.88 ± 0.00	0.91 ± 0.00	0.92 ± 0.00	0.93 ± 0.00	0.93 ± 0.00	0.94 ± 0.00	0.94 ± 0.00
ellipses	0.57 ± 0.01	0.62 ± 0.01	0.63 ± 0.01	0.64 ± 0.01	0.64 ± 0.01	0.72 ± 0.01	0.66 ± 0.01	0.75 ± 0.01	0.78 ± 0.02	0.71 ± 0.01
filter-amp	0.63 ± 0.00	0.65 ± 0.00	0.65 ± 0.01	0.67 ± 0.00	0.68 ± 0.00	0.72 ± 0.00	0.68 ± 0.00	0.71 ± 0.00	0.73 ± 0.00	0.74 ± 0.00
irregular-terms	0.88 ± 0.00	0.89 ± 0.00	0.89 ± 0.00	0.90 ± 0.00	0.92 ± 0.00	0.93 ± 0.00	0.93 ± 0.00	0.94 ± 0.00	0.95 ± 0.00	0.96 ± 0.00
islands	0.54 ± 0.01	0.55 ± 0.01	0.56 ± 0.01	0.56 ± 0.01	0.56 ± 0.01	0.55 ± 0.01	0.55 ± 0.01	0.56 ± 0.01	0.56 ± 0.01	0.56 ± 0.01
open-licensing	0.53 ± 0.00	0.54 ± 0.00	0.52 ± 0.00	0.55 ± 0.00	0.57 ± 0.00	0.55 ± 0.02	0.55 ± 0.01	0.56 ± 0.02	0.56 ± 0.02	0.56 ± 0.02
subject-verb-agreement	0.62 ± 0.00	0.63 ± 0.00	0.67 ± 0.00	0.67 ± 0.00	0.71 ± 0.00	0.75 ± 0.01	0.75 ± 0.00	0.76 ± 0.01	0.76 ± 0.01	0.76 ± 0.01
hyperonym	0.51 ± 0.00	0.56 ± 0.01	0.49 ± 0.00	0.49 ± 0.00	0.47 ± 0.00	0.51 ± 0.00	0.51 ± 0.00	0.51 ± 0.00	0.51 ± 0.00	0.51 ± 0.00
unconventional-estuary	0.47 ± 0.03	0.55 ± 0.02	0.56 ± 0.02	0.58 ± 0.02	0.60 ± 0.01	0.64 ± 0.01	0.66 ± 0.02	0.67 ± 0.01	0.68 ± 0.03	0.72 ± 0.00
subject-advocation	0.29 ± 0.00	0.32 ± 0.01	0.35 ± 0.01	0.35 ± 0.01	0.42 ± 0.02	0.46 ± 0.01	0.46 ± 0.01	0.46 ± 0.02	0.47 ± 0.01	0.47 ± 0.01
turn-taking	0.65 ± 0.02	0.66 ± 0.02	0.66 ± 0.01	0.68 ± 0.01	0.68 ± 0.01	0.71 ± 0.02	0.69 ± 0.02	0.71 ± 0.02	0.72 ± 0.01	0.73 ± 0.01
Average	0.63 ± 0.03	0.66 ± 0.03	0.67 ± 0.03	0.68 ± 0.02	0.68 ± 0.03	0.71 ± 0.03	0.69 ± 0.03	0.71 ± 0.02	0.72 ± 0.03	0.73 ± 0.02

Table 4: GLUE accuracy by task and model

Table 5: MSGS accuracy by task and model

Deep Representation Learning for Open Vocabulary Electroencephalography-to-Text Decoding

Hamza Amrani, Daniela Micucci, Paolo Napoletano

University of Milano - Bicocca, Milan, Italy

{hamza.amrani, daniela.micucci, paolo.napoletano}@unimib.it

Abstract

Previous research has demonstrated the potential of using pre-trained language models for decoding open vocabulary Electroencephalography (EEG) signals captured through a non-invasive Brain-Computer Interface (BCI). However, the impact of embedding EEG signals in the context of language models and the effect of subjectivity, remain unexplored, leading to uncertainty about the best approach to enhance decoding performance. Additionally, current evaluation metrics used to assess decoding effectiveness are predominantly syntactic and do not provide insights into the comprehensibility of the decoded output for human understanding. We present an end-to-end deep learning framework for non-invasive brain recordings that brings modern representational learning approaches to neuroscience. Our proposal introduces the following innovations: 1) an end-to-end deep learning architecture for open vocabulary EEG decoding, incorporating a subject-dependent representation learning module for raw EEG encoding, a BART language model, and a GPT-4 sentence refinement module; 2) a more comprehensive sentence-level evaluation metric based on the BERTScore; 3) an ablation study that analyses the contributions of each module within our proposal, providing valuable insights for future research. We evaluate our approach on two publicly available datasets, ZuCo v1.0 and v2.0, comprising EEG recordings of 30 subjects engaged in natural reading tasks. Our model achieves a BLEU-1 score of 42.75%, a ROUGE-1-F of 33.28%, and a BERTScore-F of 53.86%, outperforming the previous state-of-the-art methods by 3.38%, 8.43%, and 6.31%, respectively.

Introduction

The integration of deep learning into neuroscience is advancing rapidly. Over the past decades, Brain-Computer Interfaces (BCIs) have made significant improvements in decoding natural language from brain recordings to restore communication to people who have lost the ability to speak (Willett et al. 2021; Moses et al. 2021). Although effective, these approaches require invasive neurosurgery, making them difficult for most other uses.

Decoding methods that use non-invasive recordings could be more widely adopted, offering significant potential for application in both restorative and augmentative applications. Non-invasive brain recordings can capture multiple types of linguistic information (Huth et al. 2016; Broderick et al. 2018; Caucheteux and King 2022), but previous

attempts to use this information have been limited to decode sentences and words in small closed vocabularies (Pereira et al. 2018; Dash, Ferrari, and Wang 2020; Moses et al. 2021), not clarifying whether current non-invasive recordings have the spatial and temporal resolution necessary for decoding natural language. In addition, existing approaches cannot decode semantically close words.

Interestingly, previous works (Gauthier and Ivanova 2018; Caucheteux and King 2022) demonstrate that the human brain encodes language into higher-dimensional semantic representations. This is similar to how modern pre-trained language models, such as BERT (Devlin et al. 2018), BART (Lewis et al. 2019), T5 (Raffel et al. 2020), and GPT4 (OpenAI 2023), encode words into contextualized semantic embedded representations in Natural Language Processing (NLP). Thanks to their transfer learning abilities, diverse recent NLP downstream tasks, such as sequence classification, text generation, and question answering, have reached substantial improvements. Likewise, various studies (Wang and Ji 2022; Wang et al. 2023; Tang et al. 2023) experimented with combining brain signal decoding to NLP models to produce semantic brain-encoded representations. They demonstrate the ability of NLP models to extract semantic features that capture the meaning of input brain recordings.

The study by Wang et al. (Wang and Ji 2022) is the first to prove the potential of employing pre-trained language models, such as BART, to decode open vocabulary Electroencephalography (EEG) signals captured through a non-invasive Brain-Computer Interface (BCI). The processing pipeline suggested by the authors takes as input the EEG features from the ZuCo dataset (Hollenstein et al. 2018, 2019). These pre-computed EEG features are subsequently adjusted using a transformer encoder before being input into the BART model. The BART model is then fine-tuned to effectively suit the task of decoding EEG-to-Text. Recently, Duan et al. (Duan et al. 2023) present DeWave, a framework that allows for the decoding of brain dynamics into natural language without the need for eye-tracking fixations or event markers. DeWave uses a quantized variational encoder to derive discrete codex encoding and align it with a pre-trained language model. DeWave has shown superior performance compared to the state-of-the-art methods, surpassing the baseline by 3.06% and 1.9%, respectively, achieving

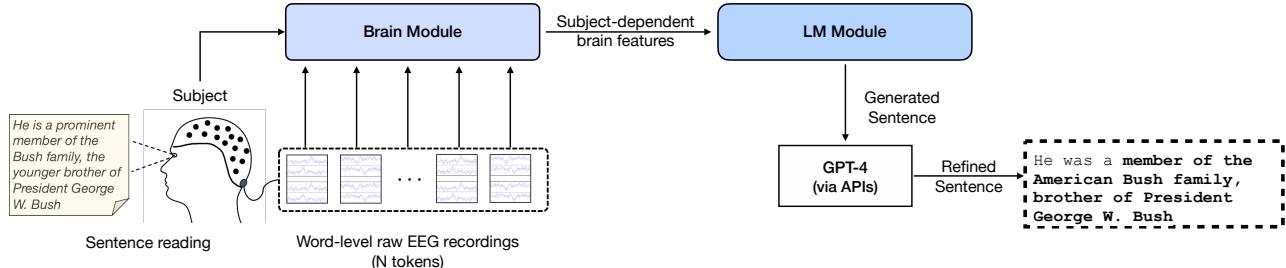


Figure 1: The workflow of the proposed method involves several steps. Firstly, the raw EEG signals corresponding to each word are input into the Brain module. This module extracts subject-dependent features, which are subsequently utilized by a Language Module based on BART suitable trained for sentence generation. The resulting sentence is further refined using GPT-4 APIs to produce the final output. In the example, the ground truth is: *He is a prominent member of the Bush family, the younger brother of President George W. Bush*; the final sentence predicted by our model is: ***He was a member of the American Bush family, brother of President George W. Bush***. Bold font refers to the exact match between the ground truth and the estimated sentence.

41.35% BLEU-1 and 30.69% Rouge-F on the ZuCo dataset.

Nevertheless, the unexplored impact of embedding EEG signals within language models raises questions about the optimal approach for enhancing decoding performance. Furthermore, while the analysis of EEG signals is a valuable way of studying brain activity, the interpretation of these signals can indeed be influenced by subjectivity (Jeng et al. 2020). A recent study by Feng et al. (Feng, Feng, and Qin 2023), on EEG-to-Text decoding task, argued that this task is considerably challenged by the EEG representation that varies with individual subjects and the text representation influenced by semantics. Lastly, current evaluation metrics that primarily focus on syntactic aspects do not adequately capture the semantics, resulting in limited comprehensibility.

In this paper, we present an end-to-end deep learning framework for non-invasive brain recordings that uses pre-trained language models for open vocabulary EEG-to-text decoding. *Firstly*, our end-to-end deep learning architecture for open vocabulary EEG decoding incorporates a representation learning module for raw EEG encoding, a language modeling module based on BART (Lewis et al. 2019), and a GPT-4 (OpenAI 2023) refinement module, enhancing the comprehensibility of the generated sentences. The representation learning module includes a subject layer, which permits taking into account the subjectivity of EEG signals, and a multi-layer transformer encoder that allows to extract latent brain representations that are then aligned into language token embeddings. *Second*, we use the BERTScore (Zhang et al. 2019) in the evaluation, which incorporates semantic judgment at the sentence level, resulting in a more comprehensive evaluation that is closer to human perception. *Thirdly*, we conducted an ablation study to analyze and distinguish the contributions of each module within our proposal, providing valuable insights for future research work.

To demonstrate the efficacy of our approach, comprehensive evaluations are conducted on two publicly available datasets, ZuCo v1.0 and v2.0 (Hollenstein et al. 2018, 2019), comprising EEG recordings from 30 subjects actively engaged in natural reading tasks. The results achieved by our

proposal, on previously unseen sentences, are a BLEU-1 score of 42.75%, a ROUGE-1-F (Lin 2004) of 33.28%, and a BERTScore-F of 53.86%, surpassing the previous state-of-the-art results by 3.38%, 8.43%, and 6.31%, respectively.

Our code is available for public access at: <https://github.com/hamzaamrani/EEG-to-Text-Decoding>

Related Work

Related work on brain-to-speech and brain-to-text decoding can be categorized into three methods by the features they are capturing: *motor imagery based*, *overt speech based*, and *inner speech based*.

Different BCI devices have been explored encompassing Electroencephalography (EEG), Electrocorticography (ECoG), and functional Magnetic Resonance Imaging (fMRI).

Motor imagery based systems, such as for instance, point-and-click (Jarosiewicz et al. 2015; Pandarinath et al. 2017; Lee et al. 2018) and imaginary handwriting (Willett et al. 2021), have high accuracy but moderately low typing rate.

Overt speech based methods for decoding or synthesizing speech show a faster communication rate. These methods require subjects to physically speak during neural recording (Anumanchipalli, Chartier, and Chang 2019; Makin, Moses, and Chang 2020) or to imagine the physical pronunciation of the sentence (Moses et al. 2021; Willett et al. 2023). These approaches make the decoding to be system language-dependent, since the same concept may have completely distinct pronunciations in different languages.

Inner speech based approaches try to address language articulation dependencies by decoding language from imagined speech and read text (Brigham and Kumar 2010; Panachakel and Ramakrishnan 2021; Wang and Ji 2022; Nieto et al. 2022; Défossez et al. 2023; Tang et al. 2023).

A major limitation for most of the approaches discussed is the constraint of using small closed vocabularies, with a low and limited number of unique words (Pereira et al. 2018; Dash, Ferrari, and Wang 2020; Moses et al. 2021).

In addition, most current approaches (Willett et al. 2021, 2023; Défossez et al. 2023) for language communication

use invasive devices (such as ECoG) or less accessible non-invasive devices (such as fMRI). This makes it challenging to collect large-scale datasets and implement approaches to help people with paralysis who can no longer speak. Nevertheless, recent studies attempt to decode inner speech by using both open vocabularies and non-invasive devices (Wang and Ji 2022; Défossez et al. 2023; Duan et al. 2023).

Our work opens the doors for similar studies of inner speech brain-to-text decoding. We investigate the representation learning of EEG signals, the inter-subject variability, the human judgment at the sentence level of generated sentences, and the use of pre-trained language models.

Method

We aim to decode neural activity from a time series of high-dimensional brain signals recorded with non-invasive electroencephalography during the natural reading of English sentences. We first define the general task of open vocabulary EEG-to-Text decoding and then introduce the proposed end-to-end architecture.

Open Vocabulary EEG-to-Text Decoding

Let's define a sequence of word-level raw EEG signals as $X \in \mathbb{R}^{C \times T}$, with C the number of EEG channels and T the number of time steps. These EEG signals are a reflection of the recorded brain activity for a specific subject denoted as s , drawn from the set S consisting of various distinct subjects. An EEG-decoding task is the task of predicting the corresponding text sentence Y in a Sequence-To-Sequence framework. Each text sentence Y is composed of English tokens $y_n \in \mathcal{V}$ from an open vocabulary \mathcal{V} . During the training phase, the EEG-subject-Text pairs can come from various subjects and various categories of reading materials.

Thus, a supervised EEG-to-Text decoding task consists in finding a decoding function $f : \{C \times T\} \times S \rightarrow \mathcal{V}$, such that f predicts Y given X and s . We denote by $\bar{Y} = f(X, s)$ the decoded/predicted text sentence from the brain signals.

Searching for f , the task is to maximize the probability of the decoded text sentence \bar{Y} :

$$p(\bar{Y}|X) = \prod_{n=1}^N p(\bar{y}_n \in \mathcal{V}|X, \bar{y}_{<n}) \quad (1)$$

where N is the length of the text sentence \bar{Y} , and \bar{y}_n is the n -th token of \bar{Y} .

Proposed Architecture

An overview of the proposed architecture is given in Figure 1 (refer to Appendix A for a detailed overview of the architecture). It is composed of two main components: 1) a Brain module that implements a representation learning approach for EEG encoding; and 2) a Language Modeling module based on BART to produce EEG-to-Text sentences and on GPT-4 for sentence-level refinement. The training process is composed of two stages. An overview of the end-to-end architecture is presented in Figure 2, where dashed boxes correspond to the modules of the architecture that undergo training, while solid boxes represent the module that

remains untrained. We start detailing the specifics of the training stages. Then we offer a more comprehensive breakdown of each module included in our architecture.

Training Stage 1 We initiate training with the Brain module: word-level EEG signals are aligned with word-tokens, as encoded by a locked, pre-trained BART Language Model, utilizing a Mean Square Error (MSE) Loss. This stage incorporates a learnable features module designed to account for EEG encoding and subjectivity. The outcome of this training stage yields EEG subject-dependent features. The alignment procedure is done by mapping the learned EEG representation Z into the BART token embeddings $BART_{enc}^{te}$, using MSE regression loss $L_{MSE}(BART_{enc}^{te}, Z)$:

$$\min_{f_{brain}} \mathcal{L}_{MSE}(BART_{enc}^{te}, f_{brain}(X)) \quad (2)$$

Training Stage 2 Moving on, the subsequent step involves fine-tuning a pre-trained Language Model based on BART, aimed at generating word sequences through the utilization of a Cross-Entropy Loss. As in Wang et al. (Wang and Ji 2022), we use the mapped embedded brain representation Z directly as initial word embeddings to feed into the pre-trained language model encoder-decoder BART (Lewis et al. 2019). The high-level idea here is that we consider each embedded EEG representation as a word-level representation, and leverage a pre-trained language model to decode to real human language (English) like traditional machine translation tasks. Then, the last hidden states from the BART decoder are fed into a multi-layer perception (MLP) to generate English tokens \bar{y}_n from the BART vocabulary \mathcal{V} .

During the training, the objective is to minimize the text reconstruction cross-entropy loss, defined as follows:

$$\mathcal{L}_{rec} = - \sum_{n=1}^N \log p(\bar{y}_n \in \mathcal{V}) \quad (3)$$

Learnable Features Module This module is included in the Brain module and it is used for extracting subject-dependent brain features from the raw EEG signals. Given a sequence of word-level raw EEG signals $X = \{x_0, x_1, \dots, x_M\} \in \mathbb{R}^{C \times T}$ and the corresponding subject $s \in S$, we first use a deep neural network f_{brain} to get the latent subject-dependent brain representation $Z = \{z_0, z_1, \dots, z_M\} = f_{brain}(X) \in \mathbb{R}$. This architecture (Figure 3) consists of (1) a learnable EEG feature block followed (2) by a subject layer to leverage inter-subject variability, which is input to (3) a multi-layer transformer encoder named *BTE* (Brain Transformer Encoder), and then to (4) a multi-layer perceptron.

The brain data is first fed to a bi-directional Gated Recurrent Unit (GRU) (Cho et al. 2014) which reads the multi-time series input in both forward and backward directions to extract learnable EEG features. The use of GRU allows to dynamically address the different lengths of word-level raw EEG signals. We then apply a fully-connected layer to the concatenated forward and backward output. Similarly to (Défossez et al. 2023), we then add a 1x1 pointwise convolution (with a kernel size of 1) without activa-

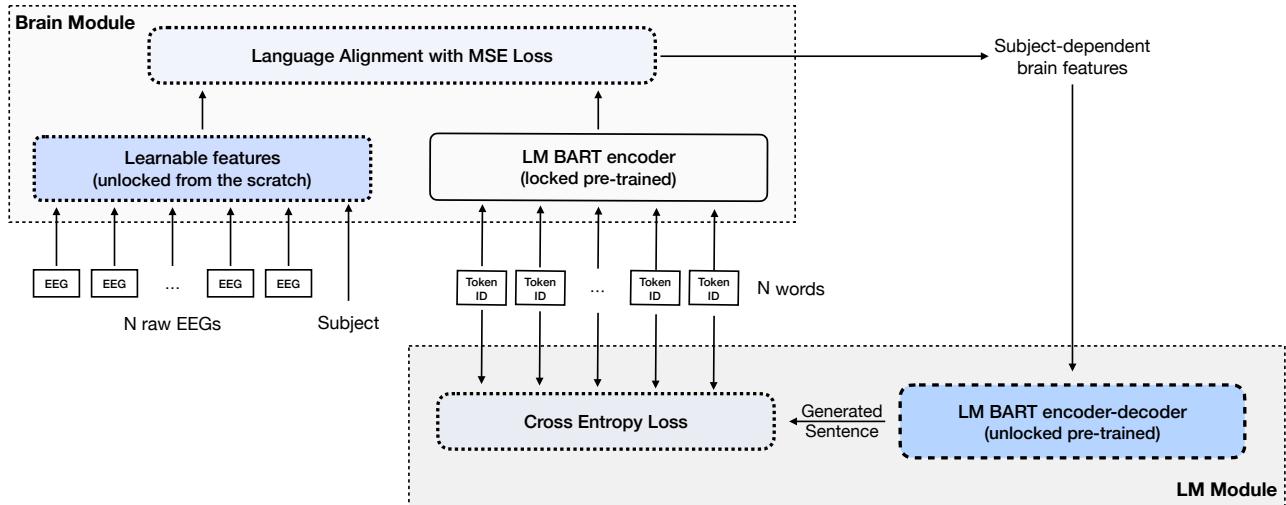


Figure 2: Overview of the proposed end-to-end architecture for open vocabulary EEG-to-Text decoding. Firstly, a sequence of word-level raw EEG signals is fed to the Brain module to extract deep-embedded representations for raw EEG encoding. Then, we use a Language Modeling (LM) module to generate EEG-to-Text sentences by leveraging the pre-trained language model BART.

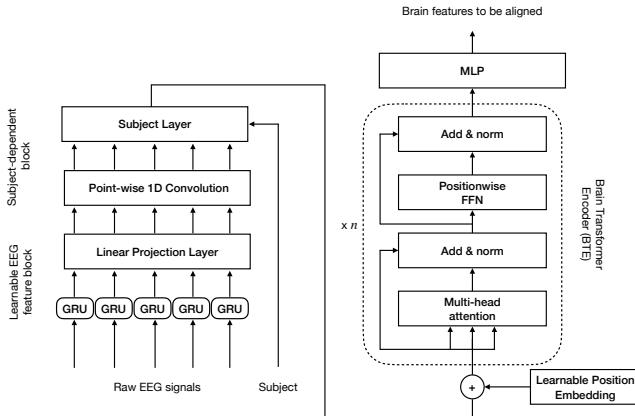


Figure 3: The Learnable features module consists of (1) a learnable EEG feature block, (2) a subject layer to leverage inter-subject variability, (3) a multi-layer transformer (Brain Transformer Encoder), and (4) an MLP.

tion and a number D of output channels. To leverage inter-subject variability, we learn a row vector $r_s \in \mathbb{R}^D$ for each subject $s \in S$ and apply it along the channel dimension. We then apply a multi-layer transformer encoder (Vaswani et al. 2017) BTE with L layers, each with H attention heads and intermediate hidden dimension d_h . The inputs to the first layer BTE_{in}^0 are produced using a weight matrix $W_{in} \in \mathbb{R}^{d_h \times l}$ and combined with a learnable 1D position embedding P (Dosovitskiy et al. 2020), which is randomly initialized. Each layer applies self-attention with causal attention masking and a feed-forward layer to the input, with layer normalization (Ba, Kiros, and Hinton 2016) and dropout (Srivastava et al. 2014) being applied after. The

outputs BTE_{out}^j of the j -th layer, become the inputs to the $(j+1)$ -th layer. Then, the final outputs BTE_{out}^L are fed into a residual MLP network, composed of two fully connected layers, obtaining the latent brain representations z_m . As we will demonstrate subsequently in the ablation study, opting to process the raw EEG signals using a recurrent neural network, rather than directly handling pre-computed features as performed by Wang et al. (Wang and Ji 2022), facilitates the extraction of subject-dependent nuances present in the brain recordings. These distinctive characteristics would otherwise remain entirely overlooked.

Sentence Refinement during Inference During the inference phase, we propose the use of the pre-trained language model GPT-4 (OpenAI 2023) via APIs on top of the generated text sentence \bar{Y} . It results in significant improvements in text comprehensibility, as well as a reduction in grammatical errors and repetitive words, enhancing the utility and effectiveness of the generated text sentence. The prompt used for the refinement is as follows:

As a text reconstructor, your task is to restore corrupted sentences to their original form while making minimum changes. You should adjust the spaces and punctuation marks as necessary. Do not introduce any additional information. If you are unable to reconstruct the text, respond with [False]. Reconstruct the following text: [text sentence \bar{Y}].

Experiments

Data

We use Zurich Cognitive Language Processing Corpus (ZuCo) (Hollenstein et al. 2018, 2019) datasets, which contain simultaneous electroencephalography and eye-tracking (ET) data recorded from natural reading tasks. The reading tasks include Normal Reading (NR) and Task-Specific

Table 1: ZuCo datasets statistics for each reading task. NR stands for Normal Reading, while TSR stands for Task-Specific-Reading.

Reading Task	#Sentences	#Train	#Val	#Test
NR v1.0	300	3,609	467	456
NR v2.0	349	2,645	343	350
TSR v1.0	407	4,456	522	601

Reading (TSR). The reading corpus of ZuCo are from movie reviews (Socher et al. 2013) and Wikipedia articles. We used data from all the subjects in ZuCo v1.0 and v2.0 (12 and 18 respectively). For the EEG recordings, high-density data were recorded at a sampling rate of 500 Hz with a band-pass of 0.1 to 100 Hz, using a 128-channel EEG Geodesic Hydrocel system (Electrical Geodesics). The recording reference was set at electrode Cz. We follow Hollenstein et al. steps (Hollenstein et al. 2018, 2019) to perform data preprocessing on raw EEG signals, leading to 105 EEG channels from the scalp recordings.

In this paper, we use concatenated sequences of word-level raw EEG signals, which were synchronized with ET fixations. We split each reading task’s data (by unique sentences) into train, validation, and test (80%, 10%, 10%), as done by Wang et al. (Wang and Ji 2022). The sentences in the test set are totally unseen. Table 1 shows the statistics of each reading task’s data. Please refer to Appendix B for a detailed description of the electrodes used.

Training Details

Architecture Details For the brain module, we set the GRU layer size to 512, and the fully connected layer to 1024. The 1d convolution maps to 64 channels and the 1d subject vector size is set to 64. The BTE has 12 layers and 8 attention heads, with an intermediate hidden dimension of 4096 and GELU activations (Hendrycks and Gimpel 2016). The last hidden states of BTE are projected on a feature space of 1024. Then, we use the large version of BART, with 12 layers for the encoder and decoder, 8 attention heads, and an intermediate hidden dimension of 4096. For GPT-4, we use OpenAI’s APIs and the model version *gpt-4*.

Optimization Settings During training, we use the SGD optimizer with a cyclical learning rate set with $5e - 7$ and $5e - 5$ as initial and upper values to update model parameters. The batch size is set to 1 during the mapping between brain and word embeddings, and then 8 during the training phase. The number of epochs is set to 25. During the training phase, we freeze the brain module weights. During inference, we use the model parameters on the best checkpoint based on the performance of the validation set.

For our architecture implementation, we use PyTorch¹ and Transformers (HuggingFace)² libraries. Both Stage1 and Stage2 were trained on a workstation equipped with

Ubuntu 22.04, 32GB RAM and 2 Nvidia GeForce GTX 1070 with 8GB Memory.

Evaluation In our experiments, we use BLEU and ROUGE metrics (Papineni et al. 2002; Lin 2004) to measure the number of words shared by two sequences. However, the lexical congruence may not fully encapsulate semantic similarity due to lexical variations denoting similar meanings. To this end, we use BERTScore (Zhang et al. 2019), an approach that uses machine learning to capture the semantic similarity between two sequences by leveraging advanced language representations derived from the BERT model (Devlin et al. 2018). BERTScore allows the integration of semantic similarity at the sentence level, leading to a more comprehensive evaluation that aligns with human perception.

Results

Improving Decoding Accuracy

We compared our architecture with the current state-of-the-art models by Wang et al. (Wang and Ji 2022) and Duan et al. (Duan et al. 2023). As shown in Table 2, our proposal achieves a BLEU-1 score of 42.75%, a ROUGE-1-F of 33.28%, and a BERTScore-F of 53.86%, showing an improvement over the state-of-the-art by 3.38%, 8.43%, and 6.31%, respectively. For larger n -grams evaluation, we obtain BLEU- $\{2,3,4\}$ scores of 25.90%, 15.66%, and 9.56% respectively, leading to an increase of 7.24%, 12.5%, and 16.30%. Our decoding embeddings resulted in higher performance for each metric, demonstrating the positive impact of learning embedded EEG representations and exploiting intersubject variability. In Appendix C we report the obtained results of our architecture for each subject. The results show a significant difference between v1.0 and v2.0 participants. On average, v2.0 participants outperform v1.0 participants by 19.64%, 42.61%, and 11.83% for BLEU-1, ROUGE1-F, and BERTScore-F respectively.

In addition to numerical results, we report decoding examples of generated EEG-to-Text sentences compared to the ground truth and the state of the art, with and without GPT-4 sentence refinement (Table 3). We observe that our model is sometimes able to precisely capture named entities that do not exist in the training set. “George W. Bush” in (1) and “Puerto Rico” in (2) are correctly decoded, while “presidential election” in (3) is incorrectly decoded. Compared to (Wang and Ji 2022), our model results in significant improvements in text comprehensibility, as well as a reduction in grammatical errors and repetitive words, as shown in example (4). Please refer to Appendix D to see additional decoding examples of generated EEG-to-Text sentences.

The complexity of open vocabulary EEG decoding tasks arises from the high dimensionality, intersubjectivity, and variability of EEG data, coupled with the intrinsic difficulties associated with the language decoding capabilities of AI-based language models. Our improvements represent significant progress in overcoming these multiple challenges and suggest a promising direction for future research in non-invasive brain decoding.

¹<https://github.com/pytorch/pytorch>

²<https://github.com/huggingface/transformers>

Table 2: Open Vocabulary EEG-to-Text decoding model evaluation on ZuCo datasets. We compare our architecture (without GPT-4 sentence refinement since it is used just on the inference phase) with the current state-of-the-art by using three distinct metrics: BLEU-N ($N = 1, 2, 3, 4$), ROUGE-1 (Precision, Recall, and F1 scores), and BERTScore (Precision, Recall, and F1 scores). We also report ablations and the hypothetical upper limit for BART with fixation words when no errors are made to map EEG signals to token words. **Bold** numbers indicate the first best result, Underline numbers indicate the second best result.

Method	BLEU-N (%) ↑			ROUGE-1 (%) ↑			BERTScore (%) ↑			
	N=1	N=2	N=3	N=4	R	P	F	P	R	F
(Wang and Ji 2022)	40.1	23.1	12.5	6.8	28.8	31.7	30.1	48.84	52.71	50.66
(Duan et al. 2023)	41.35	24.15	13.92	8.22	28.82	33.71	30.69	-	-	-
Our Architecture	42.75	25.90	15.66	9.56	30.60	36.71	33.28	52.62	55.26	53.86
w/o subject layer	<u>41.51</u>	24.41	<u>14.31</u>	8.38	<u>29.22</u>	35.40	31.92	<u>51.09</u>	<u>53.93</u>	<u>52.43</u>
w/o language alignment	41.30	<u>24.50</u>	14.14	<u>8.40</u>	29.16	<u>35.76</u>	<u>32.00</u>	50.82	53.62	52.16
w/o BTE	35.51	20.51	12.61	8.98	25.62	26.38	25.83	46.44	50.52	48.34
w/o BART finetuning	28.50	14.35	7.01	3.38	21.32	23.07	22.03	39.67	47.90	43.13
BART with fixation words	72.45	62.16	53.80	46.84	67.16	75.25	70.65	66.72	74.47	69.89

Table 3: Open Vocabulary EEG-to-Text decoding examples on ZuCo unseen test sentences. We report both predictions from our model, with and without GPT-4 sentence refinement. (1-3) are in NR v1.0, v2.0. (4) is in SR v1.0. **Bold** means exact match, *Italic* indicates semantic similarity. Underline denotes error match.

(1)	Ground truth (Wang and Ji 2022) Prediction Prediction + GPT-4	He is a prominent member of the Bush family, the younger brother of President George W. Bush... was a former member of the <i>American family</i> , and son brother of President George W. Bush ... was the member member of the <i>American family</i> . and younger brother of President George W. Bush He was a member of the <i>American Bush family</i> , brother of President George W. Bush ...
(2)	Ground truth (Wang and Ji 2022) Prediction Prediction + GPT-4	Raymond Arrieta (born March 26, 1965 in San Juan, Puerto Rico) is considered by many to be one of Puerto Rico's greatest comedians. mond wasaga,19 in 17, 18) <u>New Francisco</u> , Puerto Rico) is a one many to be the of the <i>Rico's greatest poets</i> . mond wasaga (born April 17, 1946) <u>New Francisco</u> , Puerto Rico) is a one many to be the of the <i>Rico's most artists</i> . Ramon Wasaga (born April 17, 1946 , in <u>New Francisco</u> , Puerto Rico) is one of the many to be considered as one of the most prominent artists of Puerto Rico .
(3)	Ground truth (Wang and Ji 2022) Prediction Prediction + GPT-4	Following the 1980 presidential election, Bush and his family moved to Miami-Dade County, Florida. the <u>deaths</u> election , the was <i>his wife</i> moved to <u>California</u> , Dade County, Florida the <u>wars</u> election , Bush was <i>his wife</i> moved to Florida, Dade County, Florida . <i>After the war's election</i> , Bush and <i>his wife</i> moved to Dade County, Florida .
(4)	Ground truth (Wang and Ji 2022) Prediction Prediction + GPT-4	It's not a particularly good film, but neither is it a monstrous one. was a a bad good story, but it is it bad bad . one. 's a a bad good movie , but it is it bad bad . one. It's a bad good movie , but is it a bad one.

Embedding Visualization We provide a visual comparison via t-distributed stochastic neighbor embedding (t-SNE) between the precalculated EEG features (Figure 4 (left)) as used by Wang et al. (Wang and Ji 2022), and EEG embedded representations obtained by the proposed Brain module (Figure 4 (right)). Distinct colors refer to different subjects. Each dot represents a sentence. The red triangle represents the EEG embedded representations corresponding to the same sentence “*With his interest in race cars, he formed a second company, the Henry Ford Company.*” We can observe that our learned EEG representations of sentences from the same subject are much more grouped compared with pre-calculated EEG representations, denoting the capacity of our latent space to model EEG subjectivity.

Ablations

Our ablations highlight the importance of (1) the subject layer, (2) the language alignment, (3) the use of the Brain Transformer Encoder, and (4) the BART finetuning (Table 2). First, a model trained to generate EEG-to-Text sentences without the use of the subject layer achieves lower decoding accuracy on average across datasets, that is, about 1 – 1.5% lower than our model. While modest, these scores show the positive effect of leveraging inter-subject variability. Second, we show the effect of using language alignment with MSE. The results show small differences, especially in the BLEU and ROUGE scores. For BERTScore we see small improvements. Thirdly, sentence generation without the Brain Transformer Encoder shows a significant drop in performance compared to our model. For BLEU-1 the decrease is 7.24%, while for BLEU-2 is 5.39%. While, ROUGE-1-F and BERTScore-F lose 7.45% and 5.52%, respectively.

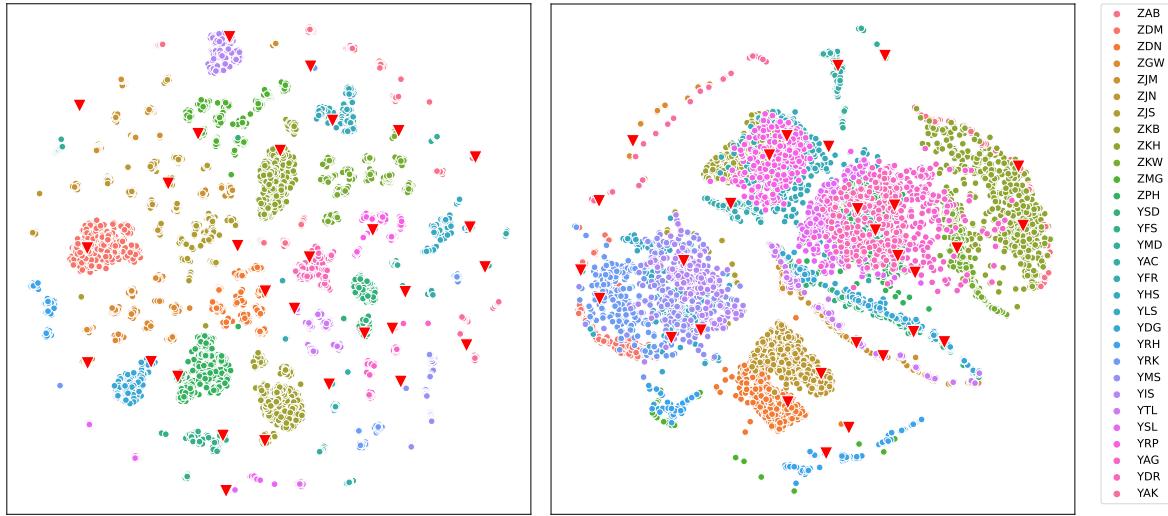


Figure 4: t-SNE visualization of EEG embedded representations of sentences in the training set, which are (a) original EEG representations and (b) generated by the Brain module of our architecture. Distinct colors mean different subjects. Each dot represents a sentence. The red triangle represents the EEG embedded representations corresponding to the same sentence "With his interest in race cars, he formed a second company, the Henry Ford Company".

We verified that the Brain Transformer Encoder provides higher decoding performances. Finally, to test whether our model effectively leverages the pre-trained BART model, we trained it without fine-tuning the BART model weights. As reported, decoding performance decreases notably up to 14.25%. This loss significantly confirms the use of fine-tuning on the BART model.

Then, we also show the hypothetical upper limit for EEG-to-Text decoding when no errors are made to map EEG signals to token words. Separately from our model, we fine-tuned BART on only Eye-Tracking fixations words without considering the raw EEG signals to reconstruct the original text sentence. It outperforms our proposed architecture by about 30% in terms of BLEU-1, 37% in terms of ROUGE-1-F, and 15% in terms of BERTScore-F. The obtained results reveal the existence of two challenges within the EEG-to-Text decoding task. The initial challenge pertains to the model's capacity to establish a dependable EEG-feature representation for the word tokens. The subsequent challenge involves the faithful reconstruction of the sentence. This experiment highlights that, between these two challenges, the foremost one is undoubtedly the ability to discern an efficacious representation of the EEG signals. This observation thereby points towards the direction of future research efforts.

Ethical Implications

While the recent advancements in utilizing brain-computer interfaces and artificial intelligence to decode neural activity into text hold significant potential in aiding individuals with communication deficits, ethical considerations, and societal impact must be carefully addressed. The scientific community must maintain vigilance and ensure that the utilization of such systems is not employed without the informed and

declared consent of the participants. Fortunately, the current nature of acquiring EEG and MEG (Magnetoencephalography) signals requires participant awareness, unlike other biomarkers such as DNA or facial features. Additionally, the susceptibility of these signals to corruption by muscle movements, such as teeth clenching or eye blinks, provides a possible precaution against unauthorized acquisition and misuse. Furthermore, it is critical to acknowledge the potential risk associated with the high subjectivity of neural signals, even in the absence of participant awareness, which could compromise mental privacy.

We strongly believe that promoting and encouraging open science practices remains essential for responsibly assessing the potential risks and benefits associated with BCI and AI technologies in this domain.

Conclusions and Future Works

In this paper, we present an end-to-end deep learning framework for open vocabulary EEG-to-Text decoding task. By leveraging a subject-dependent representation learning module, a pre-trained BART language model, and a GPT-4 sentence refinement module, this study offers a comprehensive solution that not only enhances decoding performance but also delves into the human comprehensibility of the decoded output. The incorporation of the BERTScore as an evaluation metric has enabled a more holistic assessment, capturing not only syntactic accuracy but also taking into account human understanding at the sentence level. Moreover, the conducted ablation study permitted us to understand the contribution to the proposed architecture of each component. This in-depth analysis not only validates the efficacy of each module but also provides a roadmap for further research, guiding the development of refined and optimized approaches in the future.

The empirical validation on two publicly available datasets demonstrates the effectiveness of the proposed architecture, achieving a BLEU-1 score of 42.75%, a ROUGE-1-F of 33.28%, and a BERTScore-F of 53.86%, outperforming the previous state-of-the-art results by 3.38%, 8.43%, and 6.31%, respectively. When looking at larger n -grams ratings (BLEU-2,3,4), there is an improvement of 7.24%, 12.5%, and 16.30%, respectively. Our results show that the use of raw EEG signals leads to improved results, demonstrating the effectiveness of modern representational learning approaches in neuroscience.

In summary, this research not only fills critical voids in the EEG decoding landscape but also shows the way for future investigations. By combining advanced neural network architectures with sophisticated evaluation methodologies, the study pushes the boundaries of EEG-to-text decoding and encourages continued innovation in the pursuit of more accurate and human-aligned results.

One future direction is to improve the quality of the generated embedded representations by taking into account inter-subject variability, so to increase the ability of the model to generalize across individuals. Furthermore, ethical considerations need to be at the forefront as we move forward. Ensuring privacy, establishing clear guidelines for consent, and considering the potential long-term effects of this technology on users are critical.

Acknowledgement

This work was partially funded by the National Plan for NRRP Complementary Investments (PNC, established with the decree-law 6 May 2021, n. 59, converted by law n. 101 of 2021) in the call for the funding of research initiatives for technologies and innovative trajectories in the health and care sectors (Directorial Decree n. 931 of 06-06-2022) - project n. PNC0000003 - AdvaNced Technologies for Human-centrEd Medicine (project acronym: ANTHEM). This work reflects only the authors' views and opinions, neither the Ministry for University and Research nor the European Commission can be considered responsible for them.

References

- Anumanchipalli, G. K.; Chartier, J.; and Chang, E. F. 2019. Speech synthesis from neural decoding of spoken sentences. *Nature*, 568(7753): 493–498.
- Ba, J. L.; Kiros, J. R.; and Hinton, G. E. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.
- Brigham, K.; and Kumar, B. V. 2010. Imagined speech classification with EEG signals for silent communication: a preliminary investigation into synthetic telepathy. In *2010 4th International Conference on Bioinformatics and Biomedical Engineering*, 1–4. IEEE.
- Broderick, M. P.; Anderson, A. J.; Di Liberto, G. M.; Crosse, M. J.; and Lalor, E. C. 2018. Electrophysiological correlates of semantic dissimilarity reflect the comprehension of natural, narrative speech. *Current Biology*, 28(5): 803–809.
- Caucheteux, C.; and King, J.-R. 2022. Brains and algorithms partially converge in natural language processing. *Communications biology*, 5(1): 134.
- Cho, K.; Van Merriënboer, B.; Bahdanau, D.; and Bengio, Y. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.
- Dash, D.; Ferrari, P.; and Wang, J. 2020. Decoding imagined and spoken phrases from non-invasive neural (MEG) signals. *Frontiers in neuroscience*, 14: 290.
- Défossez, A.; Caucheteux, C.; Rapin, J.; Kabeli, O.; and King, J.-R. 2023. Decoding speech perception from non-invasive brain recordings. *Nature Machine Intelligence*, 1–11.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- 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*.
- Duan, Y.; Zhou, J.; Wang, Z.; Wang, Y.-K.; and Lin, C.-T. 2023. DeWave: Discrete EEG Waves Encoding for Brain Dynamics to Text Translation. *arXiv preprint arXiv:2309.14030*.
- Feng, X.; Feng, X.; and Qin, B. 2023. Semantic-aware Contrastive Learning for Electroencephalography-to-Text Generation with Curriculum Learning. *arXiv preprint arXiv:2301.09237*.
- Gauthier, J.; and Ivanova, A. 2018. Does the brain represent words? An evaluation of brain decoding studies of language understanding. *arXiv preprint arXiv:1806.00591*.
- Hendrycks, D.; and Gimpel, K. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.
- Hollenstein, N.; Rotsztejn, J.; Troendle, M.; Pedroni, A.; Zhang, C.; and Langer, N. 2018. ZuCo, a simultaneous EEG and eye-tracking resource for natural sentence reading. *Scientific data*, 5(1): 1–13.
- Hollenstein, N.; Troendle, M.; Zhang, C.; and Langer, N. 2019. ZuCo 2.0: A dataset of physiological recordings during natural reading and annotation. *arXiv preprint arXiv:1912.00903*.
- Huth, A. G.; De Heer, W. A.; Griffiths, T. L.; Theunissen, F. E.; and Gallant, J. L. 2016. Natural speech reveals the semantic maps that tile human cerebral cortex. *Nature*, 532(7600): 453–458.
- Jarosiewicz, B.; Sarma, A. A.; Bacher, D.; Masse, N. Y.; Simeral, J. D.; Sorice, B.; Oakley, E. M.; Blabe, C.; Pandarinath, C.; Gilja, V.; et al. 2015. Virtual typing by people with tetraplegia using a self-calibrating intracortical brain-computer interface. *Science translational medicine*, 7(313): 313ra179–313ra179.
- Jeng, P.-Y.; Wei, C.-S.; Jung, T.-P.; and Wang, L.-C. 2020. Low-dimensional subject representation-based transfer learning in EEG decoding. *IEEE Journal of Biomedical and Health Informatics*, 25(6): 1915–1925.

- Lee, M.-H.; Williamson, J.; Won, D.-O.; Fazli, S.; and Lee, S.-W. 2018. A high performance spelling system based on EEG-EOG signals with visual feedback. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(7): 1443–1459.
- Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.
- Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, 74–81.
- Makin, J. G.; Moses, D. A.; and Chang, E. F. 2020. Machine translation of cortical activity to text with an encoder-decoder framework. *Nature neuroscience*, 23(4): 575–582.
- Moses, D. A.; Metzger, S. L.; Liu, J. R.; Anumanchipalli, G. K.; Makin, J. G.; Sun, P. F.; Chartier, J.; Dougherty, M. E.; Liu, P. M.; Abrams, G. M.; et al. 2021. Neuroprosthesis for decoding speech in a paralyzed person with anarthria. *New England Journal of Medicine*, 385(3): 217–227.
- Nieto, N.; Peterson, V.; Rufiner, H. L.; Kamienkowski, J. E.; and Spies, R. 2022. Thinking out loud, an open-access EEG-based BCI dataset for inner speech recognition. *Scientific Data*, 9(1): 52.
- OpenAI. 2023. GPT-4 Technical Report. *ArXiv*, abs/2303.08774.
- Panachakel, J. T.; and Ramakrishnan, A. G. 2021. Decoding covert speech from EEG—a comprehensive review. *Frontiers in Neuroscience*, 15: 392.
- Pandarinath, C.; Nuyujukian, P.; Blabe, C. H.; Sorice, B. L.; Saab, J.; Willett, F. R.; Hochberg, L. R.; Shenoy, K. V.; and Henderson, J. M. 2017. High performance communication by people with paralysis using an intracortical brain-computer interface. *Elife*, 6: e18554.
- Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 311–318.
- Pereira, F.; Lou, B.; Pritchett, B.; Ritter, S.; Gershman, S. J.; Kanwisher, N.; Botvinick, M.; and Fedorenko, E. 2018. Toward a universal decoder of linguistic meaning from brain activation. *Nature communications*, 9(1): 963.
- Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1): 5485–5551.
- Socher, R.; Perelygin, A.; Wu, J.; Chuang, J.; Manning, C. D.; Ng, A. Y.; and Potts, C. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, 1631–1642.
- Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1): 1929–1958.
- Tang, J.; LeBel, A.; Jain, S.; and Huth, A. G. 2023. Semantic reconstruction of continuous language from non-invasive brain recordings. *Nature Neuroscience*, 1–9.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.
- Wang, C.; Subramaniam, V.; Yaari, A. U.; Kreiman, G.; Katz, B.; Cases, I.; and Barbu, A. 2023. BrainBERT: Self-supervised representation learning for intracranial recordings. *arXiv preprint arXiv:2302.14367*.
- Wang, Z.; and Ji, H. 2022. Open vocabulary electroencephalography-to-text decoding and zero-shot sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 5350–5358.
- Willett, F. R.; Avansino, D. T.; Hochberg, L. R.; Henderson, J. M.; and Shenoy, K. V. 2021. High-performance brain-to-text communication via handwriting. *Nature*, 593(7858): 249–254.
- Willett, F. R.; Kunz, E. M.; Fan, C.; Avansino, D. T.; Wilson, G. H.; Choi, E. Y.; Kamdar, F.; Hochberg, L. R.; Druckmann, S.; Shenoy, K. V.; et al. 2023. A high-performance speech neuroprosthesis. *BioRxiv*, 2023–01.
- Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K. Q.; and Artzi, Y. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Appendix

A - Architecture

A detailed overview of the architecture is given in Figure 5. It is composed of two main components: 1) a Brain module that implements a representation learning approach for EEG encoding; and 2) a Language Modeling module based on BART to produce EEG-to-Text sentences and on GPT-4 for sentence-level refinement.

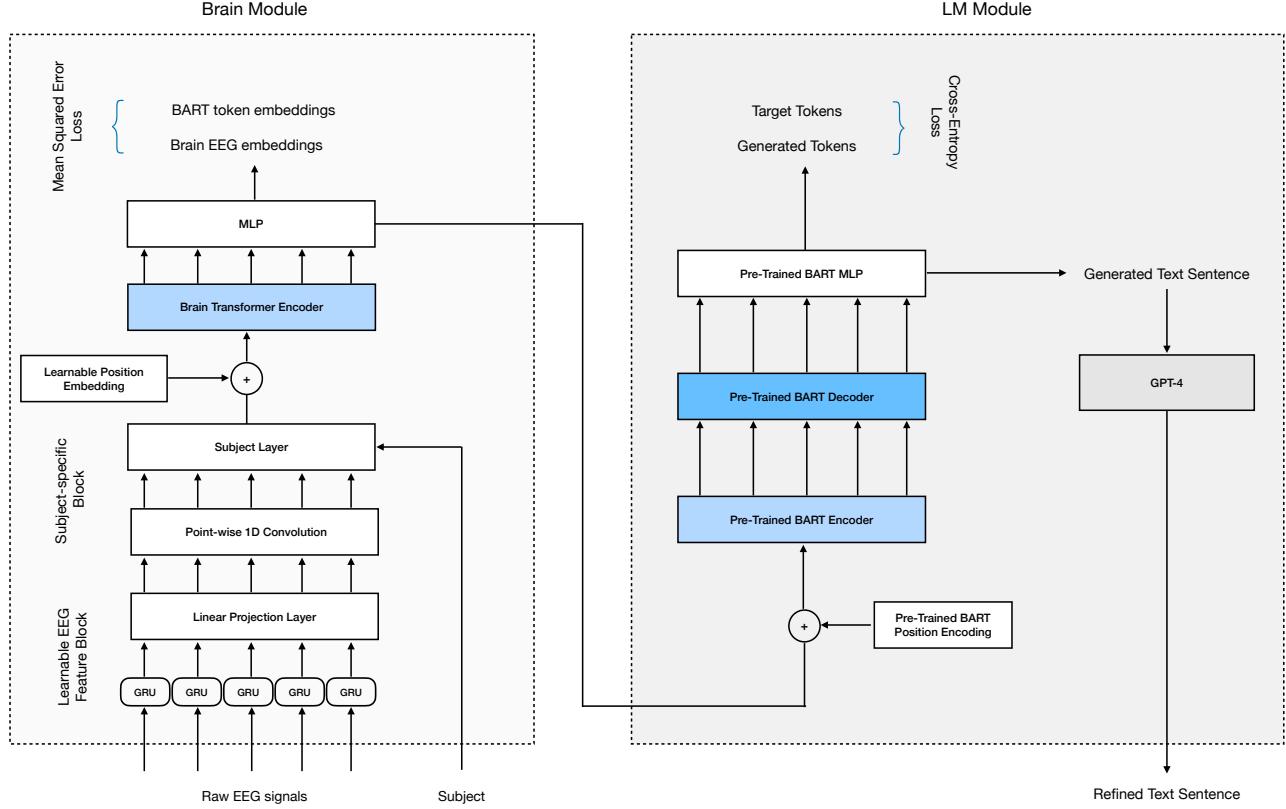


Figure 5: End-to-end architecture for open vocabulary EEG-to-Text decoding.

B - Dataset EEG electrodes

In ZuCo dataset (Hollenstein et al. 2018, 2019), we follow Hollenstein et al. steps (Hollenstein et al. 2018, 2019) to perform data pre-processing on raw EEG signals, leading to 105 EEG channels from the scalp recordings. It follows the full list of EEG channels: *E2, E3, E4, E5, E6, E7, E9, E10, E11, E12, E13, E15, E16, E18, E19, E20, E22, E23, E24, E26, E27, E28, E29, E30, E31, E33, E34, E35, E36, E37, E38, E39, E40, E41, E42, E43, E44, E45, E46, E47, E50, E51, E52, E53, E54, E55, E57, E58, E59, E60, E61, E62, E64, E65, E66, E67, E69, E70, E71, E72, E74, E75, E76, E77, E78, E79, E80, E82, E83, E84, E85, E86, E87, E89, E90, E91, E92, E93, E95, E96, E97, E98, E100, E101, E102, E103, E104, E105, E106, E108, E109, E110, E111, E112, E114, E115, E116, E117, E118, E120, E121, E122, E123, E124, Cz*.

In this paper, the Cz EEG channel has been removed as it consists of all zeros.

C - Decoding accuracy results by subject

We report open vocabulary EEG-to-Text decoding results for each subject (see Table 4). The results show a significant difference between subjects from the v1.0 and v2.0 of the dataset. The v2.0 results achieve a BLEU-1 score of 47.13%, a ROUGE-1-F of 40.16%, and a BERTScore-F of 57.35%, while the v1.0 results obtain a BLEU-1 score of 39.39%, a ROUGE-1-F of 28.16%, and a BERTScore-F of 51.28%, so leading to an increment of 19.64%, 42.61% and 11.83% respectively.

Table 4: Open Vocabulary EEG-to-Text decoding model evaluation on ZuCo datasets by each subject.

Subject	ZuCo	BLEU-N (%) ↑				ROUGE-1 (%) ↑			BERTScore (%) ↑		
		N=1	N=2	N=3	N=4	R	P	F	P	R	F
ZAB	v1.0	39.38	22.11	11.92	6.61	25.94	30.92	28.11	49.88	52.62	51.16
ZDM	v1.0	39.45	22.24	12.02	6.67	25.93	30.94	28.11	50.00	52.73	51.28
ZDN	v1.0	39.06	21.93	11.81	6.63	26.12	31.25	28.35	49.80	52.45	51.04
ZGW	v1.0	39.79	22.57	12.27	6.92	26.08	30.98	28.22	50.34	53.07	51.62
ZJM	v1.0	39.27	21.99	11.97	6.67	25.94	30.96	28.12	49.73	52.46	51.00
ZJN	v1.0	39.76	22.52	12.49	7.05	26.51	31.48	28.68	50.37	53.07	51.64
ZJS	v1.0	39.22	22.49	12.23	6.82	25.66	30.29	27.69	50.47	53.23	51.76
ZKB	v1.0	39.38	22.11	11.92	6.61	25.94	30.92	28.11	49.88	52.62	51.16
ZKH	v1.0	39.32	22.01	11.86	6.60	26.00	31.00	28.18	49.86	52.60	51.14
ZKW	v1.0	39.38	22.11	11.92	6.61	25.94	30.92	28.11	49.88	52.62	51.16
ZMG	v1.0	39.29	22.22	12.02	6.70	25.95	30.93	28.12	49.93	52.68	51.22
ZPH	v1.0	39.38	22.11	11.92	6.61	25.94	30.92	28.11	49.88	52.62	51.16
YSD	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YFS	v2.0	47.09	30.87	20.29	13.15	36.76	44.65	40.24	56.21	58.68	57.37
YMD	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YAC	v2.0	46.88	30.25	19.92	12.90	36.59	44.62	40.14	56.22	58.52	57.30
YFR	v2.0	45.82	29.23	19.09	12.21	35.91	42.64	38.91	56.51	59.13	57.74
YHS	v2.0	47.22	30.55	20.00	12.92	36.80	44.41	40.16	56.06	58.60	57.26
YLS	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YDG	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YRH	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YRK	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YMS	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YIS	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YTL	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YSL	v2.0	47.52	31.00	20.34	13.20	37.23	44.98	40.65	56.54	59.02	57.71
YRP	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YAG	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
YDR	v2.0	47.16	30.63	20.23	13.17	37.00	44.70	40.40	56.31	58.74	57.45
YAK	v2.0	47.22	30.64	20.04	12.94	36.80	44.47	40.19	56.10	58.63	57.29
Average		39.39	22.2	12.03	6.71	26.0	30.96	28.16	50.0	52.73	51.28
	v2.0	47.13	30.57	20.02	12.93	36.77	44.42	40.16	56.17	58.68	57.35
	v1.0 + v2.0	42.75	25.90	15.66	9.56	30.60	36.71	33.28	52.62	55.26	53.86

D - Decoding Examples

We report additional decoding examples of generated EEG-to-Text sentences (see Table 5), with and without GPT-4 sentence refinement. The prompt used for the GPT-4 sentence refinement is as follows:

As a text reconstructor, your task is to restore corrupted sentences to their original form while making minimum changes. You should adjust the spaces and punctuation marks as necessary. Do not introduce any additional information. If you are unable to reconstruct the text, respond with [False]. Reconstruct the following text: [text sentence \bar{Y}].

Table 5: Open Vocabulary EEG-to-Text decoding examples on ZuCo unseen test sentences, with and without GPT-4 sentence refinement.

(1)	Ground truth Prediction Prediction + GPT-4	An amateurish, quasi-improvised acting exercise shot on ugly digital video. interesting actor, un-religious improvised film performance, through a, video. Interesting actor, un-religious, improvised film performance, through a video.
(2)	Ground truth Prediction Prediction + GPT-4	Viewed as a comedy, a romance, a fairy tale, or a drama, there's nothing remotely triumphant about this motion picture. the from a kind of it satire, and love tale, and a love, it's a quite funny about it film picture. From a kind of satire, it's a love tale and quite a funny film picture about love.
(3)	Ground truth Prediction Prediction + GPT-4	It's solid and affecting and exactly as thought-provoking as it should be. 's a, well. it what it-provoking as the sounds be. Well, it's as provoking as it sounds, what a be.
(4)	Ground truth Prediction Prediction + GPT-4	It's a head-turner – thoughtfully written, beautifully read and, finally, deeply humanizing. s a greatyscinc, a to crafted, well acted, well most, a moving.. It's a great, most moving, well-crafted and well-acted scene.
(5)	Ground truth Prediction Prediction + GPT-4	"The Kid Stays in the Picture" is a great story, terrifically told by the man who wrote it but this Cliff Notes edition is a cheat. The movie "ays in the House" is a film movie about andally funny by a young who wrote it. also ish version is a little. "The movie, 'Days in the House', is a film about a young man who wrote it. It's also randomly funny. The British version is a little different.
(6)	Ground truth Prediction Prediction + GPT-4	Fans of the TV series will be disappointed, and everyone else will be slightly bored. of the film series will recognize familiar to but the will will be happy disappointed. of the film series will recognize familiar to but the will be happy disappointed.
(7)	Ground truth Prediction Prediction + GPT-4	Wedding feels a bit anachronistic alting bells like little likeachronistic, alting bells like little likeachronistic.
(8)	Ground truth Prediction Prediction + GPT-4	But what's nice is that there's a casual intelligence that permeates the script. he's most about that it's a sense, to'sates the film. He's most about that. It's a sense to states the film.
(9)	Ground truth Prediction Prediction + GPT-4	An important movie, a reminder of the power of film to move us and to make us examine our values. interesting part about but must of the importance of the to shape people. of make us think our lives. interesting part about but must of the importance of the to shape people. of make us think our lives.
(10)	Ground truth Prediction Prediction + GPT-4	Jeb Bush was born in Midland, Texas, where his father was running an oil drilling company. uan Bush was born in Newland, Texas, and his father was a a insurance company company. Juan Bush was born in Newland, Texas, and his father was an insurance company owner.

EEGFormer: Towards Transferable and Interpretable Large-Scale EEG Foundation Model

**Yuqi Chen^{1*}, Kan Ren², Kaitao Song¹, Yansen Wang¹,
Yifan Wang², Dongsheng Li¹, Lili Qiu¹**

¹ Microsoft Research ² ShanghaiTech University

yansenwang@microsoft.com renkan@shanghaitech.edu.cn

Abstract

Self-supervised learning has emerged as a highly effective approach in the fields of natural language processing and computer vision. It is also applicable to brain signals such as electroencephalography (EEG) data, given the abundance of available unlabeled data that exist in a wide spectrum of real-world medical applications ranging from seizure detection to wave analysis. The existing works leveraging self-supervised learning on EEG modeling mainly focus on pretraining upon each individual dataset corresponding to a single downstream task, which cannot leverage the power of abundant data, and they may derive sub-optimal solutions with a lack of generalization. Moreover, these methods rely on end-to-end model learning which is not easy for humans to understand. In this paper, we present a novel EEG foundation model, namely EEGFORMER, pretrained on large-scale compound EEG data. The pretrained model cannot only learn universal representations on EEG signals with adaptable performance on various downstream tasks but also provide interpretable outcomes of the useful patterns within the data. To validate the effectiveness of our model, we extensively evaluate it on various downstream tasks and assess the performance under different transfer settings. Furthermore, we demonstrate how the learned model exhibits transferable anomaly detection performance and provides valuable interpretability of the acquired patterns via self-supervised learning.

Introduction

Scalp electroencephalography (EEG) are physiological signal data that provide valuable insight into the human brain activities and has extensive applications in healthcare, e.g., disease diagnosis and medical monitoring (Lawhern et al. 2018; Tang et al. 2021, 2023; Li et al. 2023). Despite the ease of collecting EEG signals, comprehending and interpreting them often requires extensive expertise from medical professionals. To address this challenge, recent research has focused on leveraging self-supervised learning techniques to learn meaningful representations from EEG data (Yi et al. 2023; Wang et al. 2023; Li et al. 2022). These learned representations can then be fine-tuned for various downstream tasks, including seizure detection (Tang et al. 2021, 2023), abnormal detection (Darvishi-Bayazi et al. 2023), emotion recognition (Yi et al. 2023; Ye, Chen, and Zhang 2022; Song

et al. 2021; Li, Wang, and Lu 2021), etc. However, these existing works focus on pretraining upon each individual dataset corresponding to a single downstream task and fail to leverage the power of abundant data. In this paper, our primary interest lies in exploring the potential of self-supervised learning using abundant large-scale unlabeled data without human annotations.

Moreover, interpretability is a crucial concern when applying machine learning models to real-world applications (Peng et al. 2022; Ali et al. 2022; Leung et al. 2022), particularly in the healthcare community (Mendoza-Cardenas, Meek, and Brockmeier 2023; Gulamali et al. 2023). Prior research (Tang et al. 2021; Wang et al. 2023) has predominantly relied on end-to-end model learning, which poses challenges for human comprehension. Models that lack interpretability have the potential to yield unsafe and irrational outcomes, thereby increasing the risk of severe medical malpractice.

To address the above issues, we introduce EEGFORMER as a solution for large-scale EEG pretraining. Our primary objective is to investigate a discrete representation learning approach (Van Den Oord, Vinyals et al. 2017; Fortuin et al. 2018; Peng et al. 2022; Esser, Rombach, and Ommer 2021) specifically designed for EEG pretraining. We provide the evidence that the utilization of vector quantized Transformer (Vaswani et al. 2017) model can learn universal representations on EEG signals with adaptable performance on various downstream tasks compared to the conventional mask reconstruction strategy (Nie et al. 2022). Furthermore, the learned codebook, along with the discrete indices provides interpretable outcomes of the useful patterns within the data.

The contribution of the paper can be summarized as below:

- We propose a novel pretraining strategy for EEG data. EEGFORMER adopts a discrete representation learning algorithm along with reconstruction loss.
- We harness the plentiful EEG data available in the TUH Corpus (Harati et al. 2014) to construct a foundational EEG model. This marks the pioneering effort in pretraining with a massive 1.7TB EEG dataset.
- We conduct a comprehensive analysis of the pretrained foundation model EEGFORMER, evaluating its performance on four downstream corpora sourced from the TUH corpus. Additionally, we explore its transferability by applying it to the Neonate dataset (Stevenson et al.

*A preprint version of an ongoing work, conducted during Yuqi's internship at Microsoft Research. Correspondence to Kan Ren.

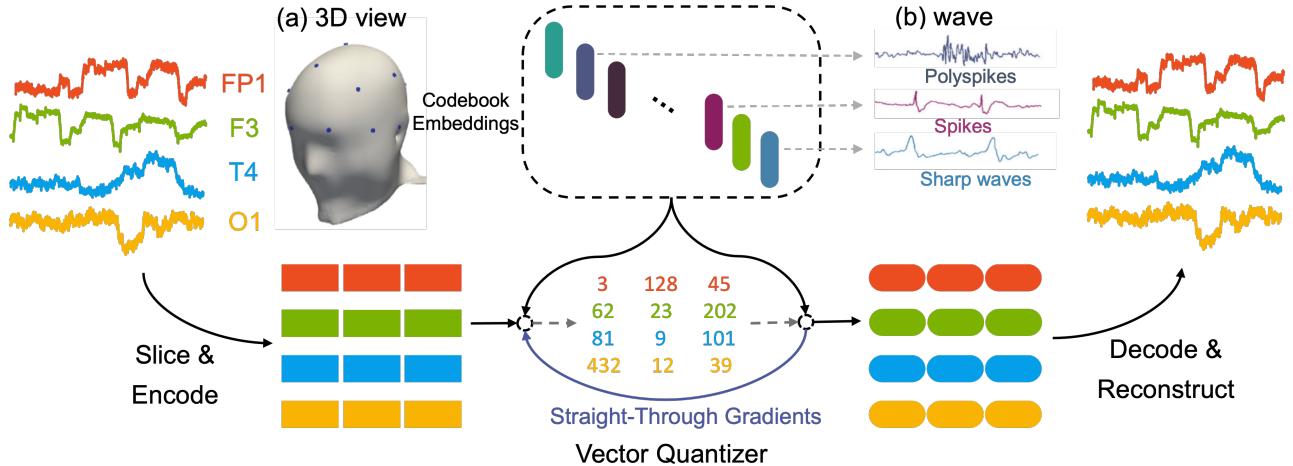


Figure 1: Overview of EEGFORMER. Initially, multi-variate EEG signals are segmented into patches, which are then passed through a Transformer encoder. Subsequently, a vector-quantized model is employed to generate discrete indices. These indices are then fed into a shallow Transformer decoder.

2019) for neonatal seizure detection.

- We provide an in-depth analysis of the learned codebook and demonstrate that the pretraining algorithm can provide transferable and interpretable representations.

Related Work

Pretraining for Time-Series Data Self-supervised learning for time-series data is a highly significant research hotspot. Many non-Transformer models have been developed to learn the representation of time series (Franceschi, Dieuleveut, and Jaggi 2019; Tonekaboni, Eytan, and Goldenberg 2021; Yue et al. 2022; Eldele et al. 2021). Recently, (Nie et al. 2022) introduced a Transformer-based approach that segments time series into patches, which leads to promising outcomes across various forecasting datasets. Furthermore, researchers are growing interested in utilizing pretrained large language models (LLMs) to enhance time series analysis (Zhou et al. 2023; Gruber et al. 2023). These methods are mainly on forecasting tasks and lack practical considerations of the model adaptation to different downstream tasks.

Pretraining for EEG data Electroencephalograms (EEGs) are widely employed for diagnosing neurological, and psychiatric, as well as in brain-machine interface applications. In the field of EEG signals, self-supervised learning has emerged as a promising approach (Tang et al. 2021; Jiang et al. 2021; Kostas, Aroca-Ouellette, and Rudzicz 2021). Seq-CLR (Mohsenvand, Izadi, and Maes 2020) introduces a set of data augmentations for EEG and extends the SimCLR (Chen et al. 2020) framework to extract channel-wise features on time-series EEG data. MMM (Yi et al. 2023) focuses on spatial and topological modeling of EEG data and breaks the boundaries between different EEG topologies. However, these methods either apply self-supervision within the same dataset or test for a single downstream task, which cannot fully unleash the power of the self-supervised pretraining

paradigm. In this paper, our approach diverges the existing methods by leveraging the extensive multiple datasets of different tasks for pretraining purposes. Furthermore, we present a novel pretraining strategy that integrates discrete representation learning, thereby enhancing interpretability.

EEGFORMER: Vector-Quantized Pretraining Transformer for EEG Data

This work aims to present a novel pretraining algorithm to derive a universal, reusable, and transferable EEG foundation model. In this paper, we focus on learning temporal patterns among multi-channel EEG data. Specifically, we view EEG data as a multi-variate time series data, i.e., $X \in \mathbb{R}^{L \times C}$, where L represents the length of the time series, and C represents the number of channels (or variates)¹. Our primary goal is to develop a self-supervised learning algorithm that optimally leverages unlabelled data while enhancing interpretability. We introduce a customized vector-quantized pre-training approach designed for EEG data to accomplish this, as illustrated in Figure 1. EEG signals can be encoded into discrete tokens, enabling interpretation through the analysis of these tokens, as is discussed in experiments. During the fine-tuning stage, the model and the codebook can be further fine-tuned to integrate specific domain-specific knowledge. In the subsequent subsections, we will provide a detailed description of the overall framework, including the preprocessing, EEG slicing, encoding module, decoding module, training algorithm, and fine-tuning processes.

Feature Preprocessing Converting EEG signals to the frequency domain is a common preprocessing technique. Following (Tang et al. 2021), given a time domain EEG signals,

¹We mitigate the sample rate discrepancy by resampling the EEG data to a uniform rate of 250 Hz. Further, our analysis focuses on fixed-length 12-second EEG data following (Tang et al. 2021).

we perform fast Fourier transformation (FFT) to obtain frequency domain amplitude as input features.

Slice & Encode To pretrain a time-series tokenizer, we first apply instance normalization to the frequency domain inputs. Then, we split each univariate time series into (non-)overlapped segments (Nie et al. 2022). Specifically, for each variate (or channel), i.e., $x_c \in \mathbb{R}^L$ for the c^{th} variate. Denote the patch length as P and the stride as S , the patching process will generate a sequence of patches $x_c \in \mathbb{R}^{P \times N}$, where $N = (\lfloor \frac{L-P}{S} \rfloor + 2)$ indicates the number of patches.

Given the input EEG data $x_c \in \mathbb{R}^{P \times N}$ for $c \in [1, \dots, C]$, it is necessary to add position embedding before input to the Transformer encoder. Specifically, we map the dimension to D via learnable weight matrix $\mathbf{w}_p \in \mathbb{R}^{P \times D}$ and adopt learnable position embedding, i.e., $\mathbf{w}_{pos} \in \mathbb{R}^{N \times D}$. Hence, the input vector is given by $\hat{x}_c = x_c^\top \mathbf{w}_p + \mathbf{w}_{pos}$. Finally, we forward \hat{x}_c into a stack of Transformer encoder layers in a channel-independent manner (Nie et al. 2022).

Vector Quantizer The vector quantizer looks up the nearest neighbor in the codebook for each patch representation h_i . Let $\{v_1, v_2, \dots, v_K\}$ denote the embeddings in the codebook. For the i^{th} patch, its quantized code is calculated as $z_i = \arg \min_j \|h_i - v_j\|_2$, where $j \in \{1, 2, \dots, K\}$. After quantizing the hidden vectors to discrete tokens, we feed the codebook embeddings $\{v_{z_i}\}_{i=1}^N$ to the decoder model.

Decode & Reconstruct The decoder model is a shallow Transformer model (Peng et al. 2022). Upon passing through the decoder model, each variate generates an output denoted as $\hat{h}_c \in \mathbb{R}^{N \times D}$. We map the outputs to the same shape as the input through $\mathbf{w}_o \in \mathbb{R}^{D \times P}$ and $\mathbf{b}_o \in \mathbb{R}^P$, i.e., $x_o = \hat{h}_c \mathbf{w}_o + \mathbf{b}_o$. Finally, we reshape the output to match the shape of X , denoted as X_{rec} .

Training Loss The training objective of EEGFORMER for each sample $X \in \mathcal{D}$ is to minimize

$$\|X_{rec} - X\|_2^2 + \sum_{i=1}^C \sum_{j=1}^N \|\text{sg}[\mathbf{H}_{i,j}] - \mathbf{v}_{Z_{i,j}}\|_2^2 - \|\mathbf{H}_{i,j} - \text{sg}[\mathbf{v}_{Z_{i,j}}]\|_2^2, \quad (1)$$

where $\text{sg}[\cdot]$ stands for the stop-gradient operator which is an identity at the forward pass while having zero gradients during the backward pass (Van Den Oord, Vinyals et al. 2017)².

Downstream Fine-Tuning To facilitate downstream fine-tuning, we utilize the pretrained model weights of both the encoder and the decoder modules. After obtaining the outputs $\hat{H} \in \mathbb{R}^{C \times N \times D}$, we feed them into the final layer for downstream tasks, such as classification or prediction. Notably, as the codebook is amenable to fine-tuning, the training objective follows a formulation akin to that of Eq. (1).

Experimental Results

Datasets Description We pretrain our model on the Temple University EEG Corpus (TUH Corpus)³, which has collected

²In Eq. (1), \mathbf{H} denotes the hidden vectors for all the variates, whereas \mathbf{h} stands for a single variate. Similarly for Z and z .

³https://isip.piconepress.com/projects/tuh_eeg/

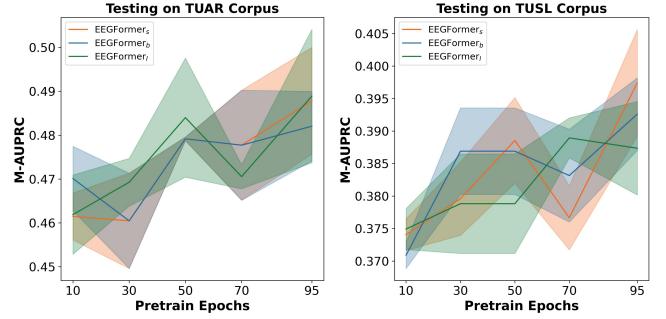


Figure 2: Influence of pretrain epochs on two TUH corpora.

over 1.7TB of unlabelled EEG data that are suitable for pre-training. We evaluate our model on five downstream datasets. i) TUAB corpus, which detects whether an EEG signal is normal or abnormal. ii) TUAR corpus, which contains annotations of 5 different artifacts. iii) TUSL corpus, which contains annotations of slowing events. v) TUSZ corpus, which contains annotations of seizure events. vi) Neonate dataset (Stevenson et al. 2019), which contains annotation of neonatal seizures. Notably, the Neonate dataset is not a subset of the TUH dataset. Therefore, we consider the transferability of our pretraining strategy.

Parameter Setting We vary the encoder layers from 6 to 12, and the codebook size, i.e., K , from 512 to 2048. The decoder is a 3-layer Transformer. We set D to 128. Specifically, EEGFORMER_s adopts a 6-layer encoder and $K = 512$, EEGFORMER_b adopts an 8-layer encoder and $K = 1024$, and EEGFORMER_l adopts a 12-layer encoder and $K = 2048$.

Compared Baselines We compare EEGFORMER with several baselines specifically for EEG data. i) EEGNet (Lawhern et al. 2018) adopts a fully convolution network for EEG data. ii) TCN (Bai, Kolter, and Koltun 2018) adopts a dilated convolutional neural network. iii) EEG-GNN (Tang et al. 2021) adopts a graph neural network for capturing spatiotemporal dependencies in EEGs. v) GraphS4mer (Tang et al. 2023) further adopts structured state space models or multivariate biosignals. Additionally, we also compare EEGFORMER with self-supervised baselines. BrainBERT (Wang et al. 2023) adopts neural signal processing techniques for producing superresolution time-frequency representations and pretrain with mask reconstruction loss⁴.

Evaluation Metrics For detection tasks, we adopt the area under the receiver operating characteristic (AUROC) and the area under the precision-recall curve (AUPRC) for evaluation. For multi-classification tasks, we adopt macro AUROC (M-AUROC) and macro AUPRC (M-AUPRC) for evaluation.

Main Results The experimental results presented in Table 1 clearly illustrate the effectiveness of our pretraining strategy in both in-dataset and transfer settings. Quantitatively, compared with the best baseline results, EEGFORMER achieves

⁴We use the pretrained weights from https://drive.google.com/file/d/14ZBOafR7RJ4A6TsurOXjFVMXiVH6Kd_Q/view.

Table 1: Experimental results on various downstream tasks. Within the table, * indicates a multi-classification task.

Model	Pretrain	Metric	TUAB	TUAR*	TUSL*	TUSZ	Neonate
EEGNet	✗	(M-)AUROC	0.841 ± .011	0.752 ± .006	0.635 ± .015	0.820 ± .030	0.793 ± .019
		(M-)AUPRC	0.832 ± .011	0.433 ± .025	0.351 ± .006	0.470 ± .017	0.499 ± .044
TCN	✗	(M-)AUROC	0.841 ± .004	0.687 ± .011	0.545 ± .009	0.817 ± .004	0.731 ± .020
		(M-)AUPRC	0.831 ± .002	0.408 ± .009	0.344 ± .001	0.383 ± .010	0.398 ± .025
EEG-GNN	✗	(M-)AUROC	0.840 ± .005	0.837 ± .022	0.721 ± .009	0.780 ± .006	0.760 ± .010
		(M-)AUPRC	0.832 ± .004	0.488 ± .015	0.381 ± .004	0.388 ± .023	0.419 ± .021
GraphS4mer	✗	(M-)AUROC	0.864 ± .006	0.833 ± .006	0.632 ± .017	0.822 ± .034	0.719 ± .007
		(M-)AUPRC	0.862 ± .008	0.461 ± .024	0.359 ± .001	0.491 ± .001	0.374 ± .013
BrainBERT	✓	(M-)AUROC	0.853 ± .002	0.753 ± .012	0.588 ± .013	0.814 ± .009	0.734 ± .019
		(M-)AUPRC	0.846 ± .003	0.350 ± .014	0.352 ± .003	0.386 ± .018	0.398 ± .027
EEGFORMER _s	✓	(M-)AUROC	0.862 ± .007	0.847 ± .013	0.683 ± .018	0.875 ± .004	0.842 ± .008
		(M-)AUPRC	0.862 ± .005	0.488 ± .012	0.397 ± .011	0.553 ± .014	0.578 ± .023
EEGFORMER _b	✓	(M-)AUROC	0.865 ± .001	0.847 ± .014	0.713 ± .010	0.878 ± .006	0.842 ± .014
		(M-)AUPRC	0.867 ± .002	0.483 ± .026	0.393 ± .003	0.560 ± .010	0.568 ± .036
EEGFORMER _l	✓	(M-)AUROC	0.876 ± .003	0.852 ± .004	0.679 ± .013	0.883 ± .005	0.833 ± .017
		(M-)AUPRC	0.872 ± .001	0.483 ± .014	0.389 ± .003	0.556 ± .008	0.544 ± .026

a 15.8% improvement on the Neonate dataset and a 14.1% on the TUSZ under the AUPRC metric.

Influence of Pretrain Epochs We conducted experiments to examine the impact of pretraining epochs on various downstream corpora. The results of these experiments are illustrated in Figure 2. Specifically, the results indicate that a longer pretraining period leads to notable enhancements in the performance of the downstream tasks.

Compared with Other Settings Table 2 compares the performance of EEGFORMER_l using fine-tuning, linear probing, and supervising from scratch. By just fine-tuning the model head (linear probing), the performance of our model is already comparable with the supervised model (GraphS4mer). Additionally, we observe that the best results are observed with end-to-end fine-tuning.

Table 2: Linear probe results on TUSL and TUAR corpus. Within the table, Sup stands for supervised learning from scratch, FT stands for self-supervised and fine-tuned, and LP stands for self-supervised and linear probing.

Model	Type	Metric	TUAR	TUSL
GraphS4mer	Sup	M-AUROC	0.833 ± .006	0.632 ± .017
		M-AUPRC	0.461 ± .024	0.359 ± .001
EEGFORMER _l	Sup	M-AUROC	0.822 ± .012	0.703 ± .033
		M-AUPRC	0.447 ± .015	0.374 ± .003
EEGFORMER _l	LP	M-AUROC	0.827 ± .000	0.657 ± .017
		M-AUPRC	0.469 ± .002	0.359 ± .003
EEGFORMER _l	FT	M-AUROC	0.852 ± .004	0.679 ± .013
		M-AUPRC	0.483 ± .014	0.389 ± .003

Towards Seizure Localization After the pertaining state, each EEG signal is discretized into multiple indices denoted

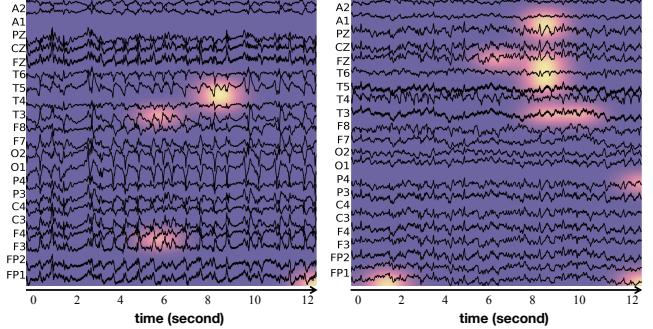


Figure 3: Interpretation results from naive Bayes model.

as $I \in [1, \dots, K]^{C \times N}$. To perform seizure detection in the TUSZ corpus using these pretrained indices, we first extract n-gram features for each data (e.g., 2-gram, 3-gram, and 4-gram). Next, we adopt a naive Bayes classifier based on n-gram features. Notably, we achieve an AUPRC of 0.292 and an AUROC of 0.741, without the need for fine-tuning the pretrained weight. Additionally, we extract the top-3 significant features with high posterior probability leading to seizure events, from the naive Bayes model. Figure 3 presents two cases, where the highlighted regions indicate the localization of seizures. It is worth noting that in the right figure, the highlighted segments correspond to the spike and slow wave complex in all the frontal lobe (Fz), parietal lobe (Pz), and temporal lobe (T3, T6), which indicates an epileptiform discharge (EPSP) followed by the refractory period of the affected neuron population after the large and synchronized neuron EPSP. This is often treated as one of the most important patterns for the diagnosis of epilepsy and the onset of a seizure event. Hence, these patterns are significant in enhancing the interpretability of the pretrained model.

Conclusion

In this paper, we introduce a novel EEG foundation model, named EEGFORMER, for self-supervised learning using large-scale EEG data. Our approach leverages a vector-quantized learning algorithm to simultaneously learn a discrete codebook and representations of multi-variate EEG signals. We extensively evaluate our pretraining algorithm on different downstream tasks using the TUH dataset, demonstrating its effectiveness. Furthermore, we perform an analysis to showcase the interpretability of our pretraining model.

References

- Ali, A.; Schnake, T.; Eberle, O.; Montavon, G.; Müller, K.-R.; and Wolf, L. 2022. XAI for transformers: Better explanations through conservative propagation. In *International Conference on Machine Learning*, 435–451. PMLR.
- Bai, S.; Kolter, J. Z.; and Koltun, V. 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*.
- Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.
- Darvishi-Bayazi, M.-J.; Ghaemi, M. S.; Lesort, T.; Arefin, M. R.; Faubert, J.; and Rish, I. 2023. Amplifying Pathological Detection in EEG Signaling Pathways through Cross-Dataset Transfer Learning. *arXiv preprint arXiv:2309.10910*.
- Eldele, E.; Ragab, M.; Chen, Z.; Wu, M.; Kwok, C. K.; Li, X.; and Guan, C. 2021. Time-series representation learning via temporal and contextual contrasting. *arXiv preprint arXiv:2106.14112*.
- Esser, P.; Rombach, R.; and Ommer, B. 2021. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12873–12883.
- Fortuin, V.; Hüser, M.; Locatello, F.; Strathmann, H.; and Rätsch, G. 2018. Som-vae: Interpretable discrete representation learning on time series. *arXiv preprint arXiv:1806.02199*.
- Franceschi, J.-Y.; Dieuleveut, A.; and Jaggi, M. 2019. Unsupervised scalable representation learning for multivariate time series. *Advances in neural information processing systems*, 32.
- Gruver, N.; Finzi, M.; Qiu, S.; and Wilson, A. G. 2023. Large language models are zero-shot time series forecasters. *arXiv preprint arXiv:2310.07820*.
- Gulamali, F. F.; Sawant, A. S.; Hofer, I.; Levin, M.; Singh, K.; Glicksberg, B. S.; and Nadkarni, G. N. 2023. Clinically Relevant Unsupervised Online Representation Learning of ICU Waveforms. In *ICLR 2023 Workshop on Time Series Representation Learning for Health*.
- Harati, A.; Lopez, S.; Obeid, I.; Picone, J.; Jacobson, M.; and Tobochnik, S. 2014. The TUH EEG CORPUS: A big data resource for automated EEG interpretation. In *2014 IEEE signal processing in medicine and biology symposium (SPMB)*, 1–5. IEEE.
- Jiang, X.; Zhao, J.; Du, B.; and Yuan, Z. 2021. Self-supervised contrastive learning for EEG-based sleep staging. In *2021 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE.
- Kostas, D.; Aroca-Ouellette, S.; and Rudzicz, F. 2021. BENDR: using transformers and a contrastive self-supervised learning task to learn from massive amounts of EEG data. *Frontiers in Human Neuroscience*, 15: 653659.
- Lawhern, V. J.; Solon, A. J.; Waytowich, N. R.; Gordon, S. M.; Hung, C. P.; and Lance, B. J. 2018. EEGNet: a compact convolutional neural network for EEG-based brain-computer interfaces. *Journal of neural engineering*, 15(5): 056013.
- Leung, K. K.; Rooke, C.; Smith, J.; Zuberi, S.; and Volkovs, M. 2022. Temporal dependencies in feature importance for time series prediction. In *The Eleventh International Conference on Learning Representations*.
- Li, R.; Wang, Y.; and Lu, B.-L. 2021. A multi-domain adaptive graph convolutional network for EEG-based emotion recognition. In *Proceedings of the 29th ACM International Conference on Multimedia*, 5565–5573.
- Li, R.; Wang, Y.; Zheng, W.-L.; and Lu, B.-L. 2022. A Multi-view Spectral-Spatial-Temporal Masked Autoencoder for Decoding Emotions with Self-supervised Learning. In *Proceedings of the 30th ACM International Conference on Multimedia*, 6–14.
- Li, Z.; Fang, Y.; Li, Y.; Ren, K.; Wang, Y.; Luo, X.; Duan, J.; Huang, C.; Li, D.; and Qiu, L. 2023. Protecting the Future: Neonatal Seizure Detection with Spatial-Temporal Modeling. *arXiv preprint arXiv:2307.05382*.
- Mendoza-Cardenas, C. H.; Meek, A.; and Brockmeier, A. J. 2023. Labeling EEG Components with a Bag of Waveforms from Learned Dictionaries. In *ICLR 2023 Workshop on Time Series Representation Learning for Health*.
- Mohsenvand, M. N.; Izadi, M. R.; and Maes, P. 2020. Contrastive representation learning for electroencephalogram classification. In *Machine Learning for Health*, 238–253. PMLR.
- Nie, Y.; Nguyen, N. H.; Sinhong, P.; and Kalagnanam, J. 2022. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*.
- Peng, Z.; Dong, L.; Bao, H.; Ye, Q.; and Wei, F. 2022. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*.
- Song, T.; Liu, S.; Zheng, W.; Zong, Y.; Cui, Z.; Li, Y.; and Zhou, X. 2021. Variational instance-adaptive graph for EEG emotion recognition. *IEEE Transactions on Affective Computing*.
- Stevenson, N. J.; Tapani, K.; Lauronen, L.; and Vanhatalo, S. 2019. A dataset of neonatal EEG recordings with seizure annotations. *Scientific data*, 6(1): 1–8.
- Tang, S.; Dunnmon, J. A.; Liangqiong, Q.; Saab, K. K.; Baykaner, T.; Lee-Messer, C.; and Rubin, D. L. 2023. Modeling Multivariate Biosignals With Graph Neural Networks and Structured State Space Models. In *Conference on Health, Inference, and Learning*, 50–71. PMLR.

Tang, S.; Dunnmon, J. A.; Saab, K.; Zhang, X.; Huang, Q.; Dubost, F.; Rubin, D. L.; and Lee-Messer, C. 2021. Self-supervised graph neural networks for improved electroencephalographic seizure analysis. *arXiv preprint arXiv:2104.08336*.

Tonekaboni, S.; Eytan, D.; and Goldenberg, A. 2021. Unsupervised representation learning for time series with temporal neighborhood coding. *arXiv preprint arXiv:2106.00750*.

Van Den Oord, A.; Vinyals, O.; et al. 2017. Neural discrete representation learning. *Advances in neural information processing systems*, 30.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention Is All You Need. *arXiv:1706.03762*.

Wang, C.; Subramaniam, V.; Yaari, A. U.; Kreiman, G.; Katz, B.; Cases, I.; and Barbu, A. 2023. BrainBERT: Self-supervised representation learning for intracranial recordings. *arXiv preprint arXiv:2302.14367*.

Ye, M.; Chen, C. P.; and Zhang, T. 2022. Hierarchical dynamic graph convolutional network with interpretability for EEG-based emotion recognition. *IEEE Transactions on Neural Networks and Learning Systems*.

Yi, K.; Wang, Y.; Ren, K.; and Li, D. 2023. Learning Topology-Agnostic EEG Representations with Geometry-Aware Modeling. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Yue, Z.; Wang, Y.; Duan, J.; Yang, T.; Huang, C.; Tong, Y.; and Xu, B. 2022. Ts2vec: Towards universal representation of time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 8980–8987.

Zhou, T.; Niu, P.; Wang, X.; Sun, L.; and Jin, R. 2023. One Fits All: Power General Time Series Analysis by Pretrained LM. *arXiv preprint arXiv:2302.11939*.

From Word Embedding to Reading Embedding Using Large Language Model, EEG and Eye-tracking

Yuhong Zhang

School of Engineering

Brown University

Providence, RI, USA

yuhong_zhang1@brown.edu

Shilai Yang

School of Engineering

Brown University

Providence, RI, USA

shilai_yang@brown.edu

Gert Cauwenberghs

Dept. of Bioengineering

UC San Diego

La Jolla, CA, USA

gcauwenberghs@ucsd.edu

Tzyy-Ping Jung

Institute for Neural Computation

UC San Diego

La Jolla, CA, USA

tpjung@ucsd.edu

Abstract—Reading comprehension, a fundamental cognitive ability essential for knowledge acquisition, is a complex skill, with a notable number of learners lacking proficiency in this domain. This study introduces innovative tasks for Brain-Computer Interface (BCI), predicting the relevance of words or tokens read by individuals to the target inference words. We use state-of-the-art Large Language Models (LLMs) to guide a new reading embedding representation in training. This representation, integrating EEG and eye-tracking biomarkers through an attention-based transformer encoder, achieved a mean 5-fold cross-validation accuracy of 68.7% across nine subjects using a balanced sample, with the highest single-subject accuracy reaching 71.2%. This study pioneers the integration of LLMs, EEG, and eye-tracking for predicting human reading comprehension at the word level. We fine-tune the pre-trained Bidirectional Encoder Representations from Transformers (BERT) model for word embedding, devoid of information about the reading tasks. Despite this absence of task-specific details, the model effortlessly attains an accuracy of 92.7%, thereby validating our findings from LLMs. This work represents a preliminary step toward developing tools to assist reading. The code and data are available in github.

Index Terms—Large Language Model, EEG, Eye-tracking, Reading assistive tools, AI for science, Natural Language Processing, Brain-Computer Interface

I. INTRODUCTION

Reading ability is fundamental for humans to acquire knowledge [1], communicate effectively [2], and is a prerequisite for expressing thoughts in writing [3]. Each word emerges as a multimodal entity and is the foundation for reading tasks. It encompasses not just semantic information accessible through word embedding models like Word2Vec [4] and BERT [5], but also the human biomarkers involved in the cognitive process.

This study used eye-tracking and electroencephalography (EEG) to assess the reading patterns of individuals in the reading tasks [6]. The internal factors of reading deficiency, such as a reader's consistent under-performance across different reading materials, may manifest in their eye-tracking patterns. These patterns can highlight issues like frequent regressions, extended fixations, a slower reading rate, challenges in information integration, and inefficient scanning [7]. Conversely,

external factors influence a reader's eye-tracking patterns in response to various text types (e.g., narrative vs. expository), the reader's prior knowledge, and interest in the subject and reading purposes [8].

Since the mid-1980s, EEG has been widely used in reading comprehension studies to explore components such as n400, n100, n1, and P2 [9]. Studies highlight the importance of EEG in improving cognitive attention and comprehension, primarily through BCI applications in reading tasks [10], [11].

Large Language Models (LLMs) can improve readers' comprehension through contextual analysis, inference skills, multilingual capabilities, question generation, and answering abilities [12]. Recent studies showed a consistency between LLMs and human subjects in reading comprehension. Specifically, subjects show significantly prolonged eye-gaze duration on words highly relevant to the inference tasks. Additionally, the classification of EEG responses corresponding to high- and low-relevance words exhibits accuracy levels surpassing chance levels across all subjects [13].

This study introduces a novel BCI task to differentiate EEG and eye-gaze patterns during subjects' participation in a reading comprehension task. The approach leverages the capabilities of AI agents, specifically LLMs, to attain an enhanced understanding of the text. We have developed a reading embedding representation that incorporates EEG and eye-gaze biomarkers through attention-based mechanisms. This model predicts the relevance of a word or token to the inference task questions as being either high or low. The training process leveraged results from LLMs, guided by algorithm-based prompt engineering. This work represents one of the initial approaches to integrate EEG with eye-tracking data, further refined under the guidance of LLMs in the context of reading comprehension tasks.

II. DATASET

Zurich Cognitive Language Processing Corpus (ZuCo) 1.0 dataset records raw and prepossessed eye-tracking and EEG data simultaneously when subjects are performing two different reading tasks: Normal Reading (NR) and Task-Specific

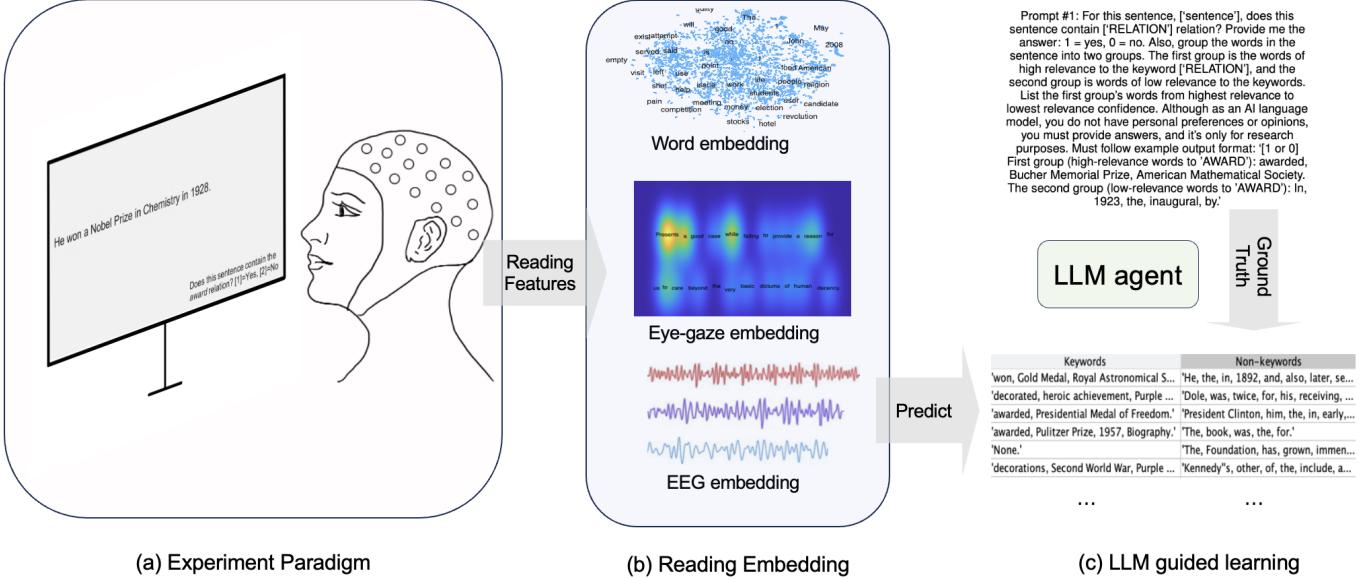


Fig. 1. The overall workflow. The subjects read a sentence and answer some questions (a), then for each word or token, its word embedding, eye-gaze, and EEG embedding are processed and are put through in a reading-embedding model (b). The model is trained under the guidance of LLM, which produces the fuzzy ground truth labels (c).

Reading (TSR), shown in Fig. 1 (a). Pre-processed EEG and eye-gaze data are segmented at the word level. In this study, we use ZuCo 1.0 task 3 TSR for our reading embedding development as subjects have the highest average accuracy among all three tasks [14]. We excluded “VISITED” relation in Task 3 for clarity, thus we have 8 inference target words: “AWARD”, “EDUCATION”, “EMPLOYER”, “FOUNDER”, “JOB TITLE”, “NATIONALITY”, “POLITICAL AFFILIATION”, “WIFE”. We excluded the data from 3 subjects because they missed at least one of the 8 word relations. We used data from nine subjects in total.

III. METHOD

A. Word Embedding

Word embeddings are fundamental components in NLP tasks. We use BERT to tokenize sentences and transform each token into high-dimensional vectors. During the tokenization process, the pre-trained BERT model skips minor tokens, such as punctuation and year numbers. Unlike non-contextual embeddings such as Word2Vec, BERT encodes the semantic meaning of words by considering the full context, given its bidirectional transformer architecture. The cosine similarity between two vectors describes their semantic closeness. We use the hidden state from the second to last layer, the eleventh out of twelve output layers, to represent each token. We then apply L2 normalization to unify each vector in the hidden state. For each word ω , the embedding dimension of BERT is 768. Consequently, we obtain a tensor of dimensions $[N \times M \times 768]$ for word embedding, where N represents the number of sentences, and M denotes the maximum number of words

in a sentence. To address sentences with fewer words, we pad them with zeros to ensure consistency.

B. Eye-gaze and EEG Embedding

We extract 12 distinct eye-gaze features from the processed ZuCo dataset. These include the Number of Fixations, Mean Pupil Size, First Fixation Duration (FFD), Total Reading Time (TRT), Gaze Duration (GD), Go-Past Time (GPT), Single Fixation Duration (SFD), and Pupil Size for FFD, TRT, GD, GPT, SFD. Each feature independently represents the subject’s reading attention but cannot capture their behavioral pattern holistically. We apply L1 normalization to each of the 12 eye features within each sentence dimension (Fig. 2).

The conditional entropy method is widely used to analyze EEG data for feature extraction [13]. It quantifies the information or uncertainty in one EEG signal given the knowledge of another. After flattening the upper triangular conditional matrix, the EEG feature dimension is 5460. For each word, there may be cases where no fixation occurs, resulting in absences of corresponding eye-gaze and EEG data. In such instances, we assign zero vectors to the respective word or token cases. For words with more than one fixation, we apply the L2 norm to each vector and take their element-wise addition (Fig. 2).

C. Prompt Engineering

We treat LLMs as independent learning agents and input the same sentence corpus and reading questions to assess their understanding abilities in both Tasks 1 and 3 of ZuCo 1.0. Twelve subjects achieved mean accuracy of 79.53% and 93.16% for Tasks 1 and 3, respectively. In comparison, GPT-3.5 Turbo achieved 93.74% and 95.59%, while GPT-4 reached

97.44% and 98.82% for Tasks 1 and 3, respectively, all surpassing human performances in sentence understanding [13].

For each word, we assign labels l to denote whether it belongs to the group that is highly or lowly relevant to one of the eight target inference words, we denote the labels as HRW and LRW. The labels are created through a joint selection process involving GPT-3.5 Turbo and GPT-4, using two prompts to generate the labels, as shown in Fig. 1 (c). The detailed prompting process is described in [13].

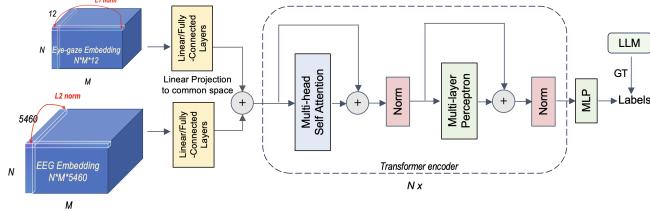


Fig. 2. Training data and the Reading-Embedding Model.

D. Reading embedding

We engineered and evaluated a novel Reading-Embedding Model. This model can encode biomarker vectors derived from EEG, eye-tracking, or a combination of both, using a single transformer encoder block equipped with multi-headed self-attention. We use the same architecture to map word embeddings to their respective labels obtained by LLMs.

We avoid concatenating the two embeddings directly to address the problem of significant dimensional disparity. Instead, we linearly project each feature into a common space of dimension 128. For EEG and eye-gaze features, we combine them using element-wise addition. These projected features are then further processed using sinusoidal positional encoding before being put through the transformer encoder. The transformer encoder is then followed by a Multi-Layer Perceptron (MLP) to output probabilities of input samples to be of a specific label in this binary classification task (Fig. 2).

We compose the total loss for the model as a weighted summation of 1) Masked Binary Cross Entropy Loss. 2) Masked Mean Squared Error Loss 3) Masked Soft F1 Loss (based on Bray-Curtis Dissimilarity [15])

$$\mathcal{L}_{tot} = \lambda_1 \mathcal{L}_{bce} + \lambda_2 \mathcal{L}_{mse} + \lambda_3 \mathcal{L}_{f1} \quad (1)$$

For simplicity, the weights $\lambda_i, i = 1, 2, 3$ for the loss terms are set constant in our experiments, though they could be set as trainable variables [16]. The three loss terms can be calculated as:

$$\mathcal{L}_{bce} = -\frac{1}{N} \sum_{i=1}^N \mathbb{1}_i \cdot [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)] \quad (2)$$

$$\mathcal{L}_{mse} = \frac{1}{N} \sum_{i=1}^N \mathbb{1}_i (y_i - p_i)^2 \quad (3)$$

$$\mathcal{L}_{f1} = 1 - \frac{\sum_{i=1}^N \mathbb{1}_i \cdot y_i \cdot p_i}{\sum_{i=1}^N \mathbb{1}_i \cdot y_i + \sum_{j=1}^N \mathbb{1}_j \cdot p_j} \quad (4)$$

where, N is the number of samples, $\mathbb{1}_i$ is the indicator (mask) for each sample (with $\mathbb{1}_i = 1$ for valid samples and $\mathbb{1}_i = 0$ for samples to be ignored), y_i is the true label of the i -th sample, and p_i is the predicted probability for the i -th sample to be of label HRW.

To prevent the LRW samples from dominating performance evaluations, we downsample the LRW samples to align with the number of HRW samples for both training and testing. We assess the model's performance in this binary classification task through 5-fold cross-validation applied to data from each experimental subject. For optimization, we use Stochastic Gradient Descent (SGD) with a learning rate of 0.05.

IV. RESULTS ANALYSIS

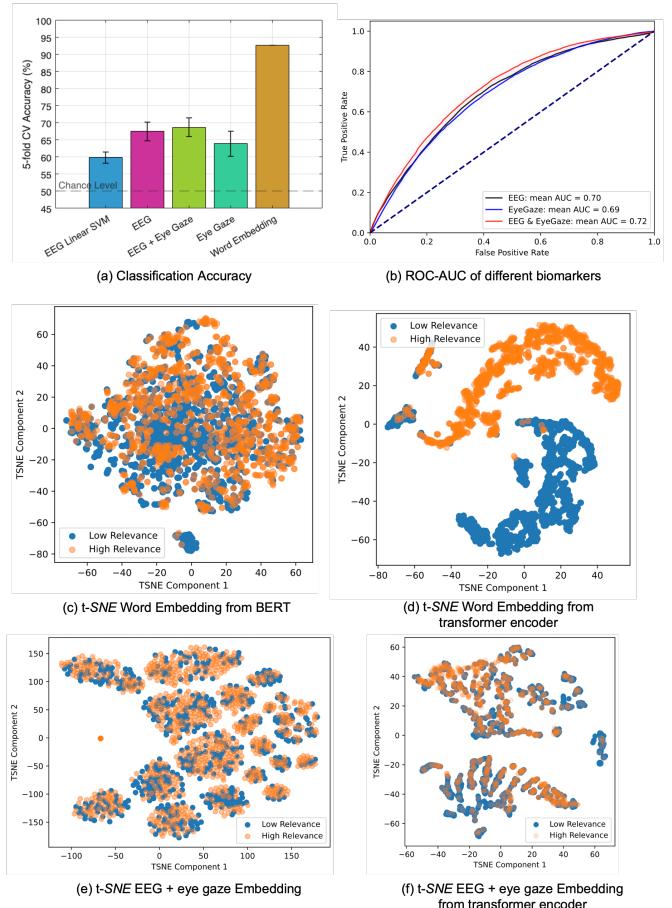


Fig. 3. Classification results and t-SNE visualization.

Fig. 3 (a) shows the binary classification results for nine subjects on word embedding, EEG and eye-gaze. When processed through a linear SVM as the classifier, EEG achieves moderate accuracy slightly above chance level, with the transformer classifier showing a noticeable advantage over SVM with an average accuracy of 67.5% and a maximum of 70.0%. Integrating EEG and eye-gaze data enhances accuracies for both classifiers, emphasizing the benefits of multi-modal approaches, with an average of 68.7% and a maximum of 71.2% in a single subject's (ZAB) data. Eye-gaze data alone

yields comparable accuracy results to EEG with a linear SVM classifier, surpassing the chance level with averages of 63.9% and 59.8%, respectively. Word embeddings from the pre-trained BERT model achieve superior performance with a 92.7% accuracy using the transformer classifier. Fig. 3 (b) illustrates the Receiver Operating Characteristic (ROC) curves for three classifiers using EEG, eye gaze data, and a combination of both. The the results using the combined EEG and EyeGaze data yield a marginally higher mean Area Under the Curve (AUC) of 0.72, showing a slightly better overall performance in classification tasks compared to using EEG (mean AUC = 0.70) or EyeGaze data (mean AUC = 0.69) individually. Fig. 3 (c-f) show t-SNE visualizations contrasting the clustering of low- and high-relevance word embeddings from BERT that are put through before and after a transformer encoder, as well as the integrated EEG and eye-gaze features. Clearly, the word embedding patterns are significantly more distinct than biomarker embeddings, addressing the difficulties of biomarkers classification.

V. DISCUSSION AND CONCLUSION

This study analyzed three distinct information modalities during reading comprehension tasks: word embedding, EEG and eye-gaze. It is not surprising that word embeddings achieved the highest accuracy because of its inherent language model attributes, albeit not being a large model. Nevertheless, fine-tuning the word embedding to classify High- and Low-relevance words is valuable for several reasons.

First, we previously used prompts developed with GPT-3.5's and GPT-4's APIs to categorize words into HRW and LRW. Although we validated word results through joint selection from both LLMs, the process remained opaque. Note that while the labels from LLMs represent the relevance degree of actual reading inference tasks, the pre-trained BERT does not inherently contain this information. Instead, it encodes words into vector forms within the context of a sentence, representing semantic similarities. Although categorizing results from BERT correlate highly with those from GPTs, they are not identical. The quantifiable representations of words render the word grouping process explainable and validate our prompt-based approach. Additionally, the accuracy using the word embeddings from BERT provides an indirect assessment of the efficacy of prompts and the overall performance of LLMs.

However, relying solely on word embedding performance is inadequate for assessing subjects' reading patterns. Instead, EEG and eye-tracking biomarkers provide valuable insights into the reading behavior of each individual. Recognizing the low signal-to-noise ratio (SNR) inherent in bio-signals, we applied feature engineering to the raw EEG and eye-gaze datasets. This process was followed by normalization and the use of linear layers to project the data into a common space, addressing dimension incompatibility. Subsequently, the two information modalities were combined and then encoded using a transformer encoder. Element-wise addition innovatively represents word's reading attributes, taking into account multiple fixations per word. A higher summed value signifies greater

attention, making it more intuitive than direct concatenation. The results obtained from EEG, eye-gaze, and their combined embedding, ranging between 65-71%, showcasing a notable enhancement compared to the 58-62% achieved with SVM in our prior study. This highlights the superior performance of transformers over conventional statistical learning models.

Given the findings, we are optimistic about the prospects of developing novel reading assistive tools that leverage AI agents, particularly integrating LLMs and multi-modality approaches in BCI applications. Future work includes applying the Reading Embedding to tasks with lower reading performance, where subjects might confuse LRWs with HRWs. Such mistakes may correlate with a decline in performance, a scenario where assistive tools can come into play.

REFERENCES

- [1] E. D. Hirsch, "Reading comprehension requires knowledge of words and the world," *American educator*, vol. 27, no. 1, pp. 10–13, 2003.
- [2] J. Ricketts, "Research review: Reading comprehension in developmental disorders of language and communication," *Journal of Child Psychology and Psychiatry*, vol. 52, no. 11, pp. 1111–1123, 2011.
- [3] L. K. Allen, E. L. Snow, S. A. Crossley, G. Tanner Jackson, and D. S. McNamara, "Reading comprehension components and their relation to writing," *L'Année psychologique*, vol. 114, no. 4, pp. 663–691, 2014.
- [4] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," 2013.
- [5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
- [6] P. C. Gordon, R. Hendrick, M. Johnson, and Y. Lee, "Similarity-based interference during language comprehension: Evidence from eye tracking during reading," *Journal of experimental psychology: Learning, Memory, and Cognition*, vol. 32, no. 6, p. 1304, 2006.
- [7] A. C. Gran Ekstrand, M. Nilsson Benfatto, and G. Öqvist Seimyr, "Screening for reading difficulties: Comparing eye tracking outcomes to neuropsychological assessments," in *Frontiers in Education*, vol. 6. Frontiers Media SA, 2021, p. 643232.
- [8] R. Smith, P. Snow, T. Serry, and L. Hammond, "The role of background knowledge in reading comprehension: A critical review," *Reading Psychology*, vol. 42, no. 3, pp. 214–240, 2021.
- [9] M. Kutas and S. A. Hillyard, "Reading senseless sentences: Brain potentials reflect semantic incongruity," *Science*, vol. 207, no. 4427, pp. 203–205, 1980.
- [10] X. Liu and Z. Cao, "Enhance reading comprehension from eeg-based brain-computer interface," in *Australasian Joint Conference on Artificial Intelligence*. Springer, 2023, pp. 545–555.
- [11] X. Gu, Z. Cao, A. Jolfaei, P. Xu, D. Wu, T.-P. Jung, and C.-T. Lin, "Eeg-based brain-computer interfaces (bcis): A survey of recent studies on signal sensing technologies and computational intelligence approaches and their applications," *IEEE/ACM transactions on computational biology and bioinformatics*, vol. 18, no. 5, pp. 1645–1666, 2021.
- [12] S. Bubeck, V. Chandrasekaran, R. Eldan, J. Gehrke, E. Horvitz, E. Kamar, P. Lee, Y. T. Lee, Y. Li, S. Lundberg, H. Nori, H. Palangi, M. T. Ribeiro, and Y. Zhang, "Sparks of artificial general intelligence: Early experiments with gpt-4," 2023.
- [13] Y. Zhang, Q. Li, S. Nahata, T. Jamal, S. kuen Cheng, G. Cauwenberghs, and T.-P. Jung, "Integrating ilm, eeg, and eye-tracking biomarker analysis for word-level neural state classification in semantic inference reading comprehension," 2023.
- [14] N. Hollenstein, J. Rotsztejn, M. Troendle, A. Pedroni, C. Zhang, and N. Langer, "Zuco, a simultaneous eeg and eye-tracking resource for natural sentence reading," *Scientific data*, vol. 5, no. 1, pp. 1–13, 2018.
- [15] J. T. Bray, J. Roger, Curtis, "An ordination of the upland forest communities of southern wisconsin," *Ecological Monographs*, vol. 27, no. 4, pp. 325–349, 1957.
- [16] L. McClenny and U. Braga-Neto, "Self-adaptive physics-informed neural networks using a soft attention mechanism," 2022.

Contextual Feature Extraction Hierarchies Converge in Large Language Models and the Brain

Gavin Mischler*

Department of Electrical Engineering
Columbia University
New York, NY 10027
gm2944@columbia.edu

Yinghao Aaron Li*

Department of Electrical Engineering
Columbia University
New York, NY 10027
y14579@columbia.edu

Stephan Bickel

The Feinstein Institutes for Medical Research
Northwell Health
Manhasset, NY 11030
sbickel@northwell.edu

Ashesh D. Mehta

The Feinstein Institutes for Medical Research
Northwell Health
Manhasset, NY 11030
amehta@northwell.edu

Nima Mesgarani[†]

Department of Electrical Engineering
Columbia University
New York, NY 10027
nima@ee.columbia.edu

Abstract

Recent advancements in artificial intelligence have sparked interest in the parallels between large language models (LLMs) and human neural processing, particularly in language comprehension. While prior research has established similarities in the representation of LLMs and the brain, the underlying computational principles that cause this convergence, especially in the context of evolving LLMs, remain elusive. Here, we examined a diverse selection of high-performance LLMs with similar parameter sizes to investigate the factors contributing to their alignment with the brain's language processing mechanisms. We find that as LLMs achieve higher performance on benchmark tasks, they not only become more brain-like as measured by higher performance when predicting neural responses from LLM embeddings, but also their hierarchical feature extraction pathways map more closely onto the brain's while using fewer layers to do the same encoding. We also compare the feature extraction pathways of the LLMs to each other and identify new ways in which high-performing models have converged toward similar hierarchical processing mechanisms. Finally, we show the importance of contextual information in improving model performance and brain similarity. Our findings reveal the converging aspects of language processing in the brain and large language models and offer new directions for developing LLMs that align more closely with human cognitive processing.

*These authors contributed equally to this work

[†]Corresponding author

1 Introduction

The intersection of artificial intelligence and neuroscience has emerged as a frontier of great interest, particularly in understanding how large language models (LLMs) and the human brain process language. Prior research has laid foundational work in this area, uncovering intriguing parallels in feature extraction and representational similarities between LLMs and neural responses during language processing. Studies [1, 2, 3, 4, 5, 6, 7, 8, 9] have demonstrated that the representations learned by LLMs can be linearly mapped to neural responses, suggesting that both LLMs and the brain utilize comparable features in language processing. However, these findings offer limited insight into the fundamental characteristics of LLMs that enable this brain-like processing.

Further investigations have delved into different aspects of LLMs to elucidate their resemblance to brain processes. Some studies [10, 11] support the predictive coding hypothesis in human language processing by finding stronger similarities with autoregressive LLMs. Others [8, 5, 12, 13] have explored various factors, such as the LLM language modeling performance, model size and capacity, and the generalizability of linguistic representations as indicators of brain-like processing. These studies imply that the quality of an LLM significantly contributes to its brain-like representations, yet the underlying reason for this similarity remain an open question. Is it merely a matter of scaling up the models [12], or do these models share fundamental computational principles that increasingly align well with the spoken language processing pathway in the human brain? This question is significant as it may suggest a potential shift in the paradigm of model optimization. Although both brains [14, 15, 16, 17] and LLMs [18, 19] process speech and language in hierarchical pathways, most studies have analyzed the similarity of their representations without detailed comparisons of the hierarchical processes through which they are created. Thus, it is still unclear whether brains and models arrive at similar representations through the same or different pathways.

We aim to answer these questions by examining the interplay between LLM performance, neural predictability, anatomical alignment, and contextual encoding, potentially paving the way toward models that perform with high accuracy and process language in a manner similar to the human brain. We examined 12 open-source, pre-trained LLMs, all uniform in size but diverse in their linguistic capabilities, particularly in language-understanding tasks such as reading comprehension. We recorded neural responses with intracranial electroencephalography (iEEG) in the auditory cortex and speech processing regions of neurosurgical patients as they listened to speech. We then predicted these neural responses from the embeddings extracted from each layer of the LLMs as they processed the same linguistic input. This approach allowed us to pinpoint which layers and aspects of the LLMs were most predictive of brain activity and explore how variations in model performance align with differences in neural prediction and their anatomical and functional correspondence. Our findings offer a fresh perspective on the evolving landscape of LLMs, providing insights that reveal more intricate parallels in language comprehension between artificial and biological systems, uncover new potential reasons for LLM performance differences, and point to a convergence in LLMs towards a more optimal, brain-like language processing system.

2 Results

2.1 Brain Similarity of Large Language Models

We studied 12 recent, popular, open-source LLMs, all with approximately 7 billion parameters. We evaluated each model on a suite of benchmark tasks to assess its language modeling performance, splitting these tasks into categories relevant to English language comprehension, specifically reading comprehension and commonsense reasoning as in [20] (see Methods for details). Overall LLM performance was estimated as the average score over these two categories.

Neural responses were recorded with invasive electrodes (intracranial EEG) from eight neurosurgical patients, with electrode placement determined by clinical need (Supplementary Fig. 1). The subjects listened to between 20 and 30 minutes of speech from various talkers, including stories voiced by voice actors and dialogues between characters. The text for each audio was fed into each LLM, and we extracted the causal embeddings of each word at every layer. We reduced these embeddings to 500 components with PCA, ensuring consistent dimensionality across models since the models were only approximately the same size to begin with. We used ridge regression to estimate the similarity of a model’s features to the brain (Fig. 1) [10, 3, 8]. We analyzed 707 electrodes which were responsive

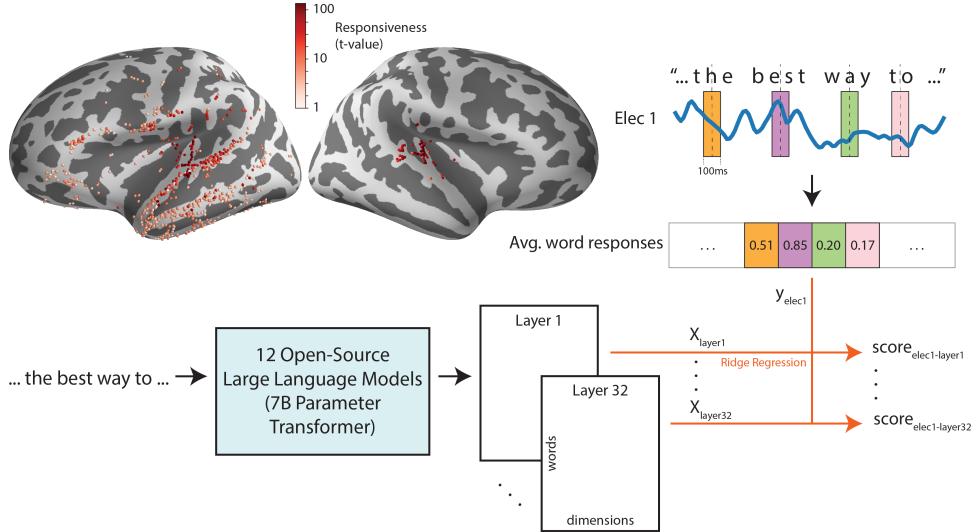


Figure 1: Mapping LLM embeddings to the brain. Speech responsive electrodes are shown on an inflated brain (shaded by their responsiveness t-value from a paired t-test between speech and silence). As subjects listened to speech, the average neural response in a 100ms window around a word center was used as a given electrode’s word response. The same text was fed to an LLM and the embeddings from all 32 layers were extracted. Ridge regression was used to predict the word responses from the LLM representations, producing a brain correlation score for each electrode-layer pair.

to speech, as determined by a t-test between responses to words and silence (FDR corrected, $p < 0.05$ [21]). For each responsive electrode, we extracted the average high-gamma band envelope response in a 100ms window around the center of every word. Then, we fit cross-validated ridge regression models to predict these neural responses from the word embeddings and used the average prediction correlation on the withheld folds as the brain similarity with that electrode. Neither the number of principal components of the embeddings nor the window size used to compute the neural response to words significantly impacted the results (Supplementary Fig. 2).

Electrode-averaged brain similarity over each model’s layers is shown in Fig. 2A. With these latest LLMs, we confirm previous findings showing that neural responses can be predicted from model representations, and we find that brain similarity generally increases over layers and peaks in middle or later layers [3, 8]. Higher-performing LLMs also achieve higher peak brain scores (Pearson $r = 0.92, p = 2.24 \times 10^{-5}$) (Fig. 2B), indicating that they extract more brain-like features from language.

Similar to the layers of a model, the auditory and language processing pathway demonstrates hierarchical organization [14, 22, 15, 16]. The primary auditory cortex, the first point of auditory processing in the cortex, is centered around posteromedial Heschl’s gyrus (pmHG, or TE1.1) [23]. Since this is a common reference point in auditory cortical processing, we quantify the depth of each electrode in the brain’s spoken language processing pathway using its distance from this landmark [24, 25, 26]. Prior studies have found that deeper layers of LLMs correspond better to deeper language processing regions of the brain [8, 11, 27]. We confirm this result (Fig. 2C), but interestingly, we also find that better-performing LLMs peak in brain similarity at earlier layers compared to worse models (Pearson $r = -0.81, p = 0.0013$) (Fig. 2D). This uncovers a new dimension in the evolution of LLMs: the progression of feature extraction over layers aligns differently with the brain for higher-performing versus lower-performing models.

2.2 Alignment of Language Processing Hierarchies Between Models and the Brain

Given that the layer-wise brain similarity appears different between good and bad models, we hypothesized that better models were not only learning more brain-like features, but that the progression of feature extraction within these models was different. Taking inspiration from an investigation

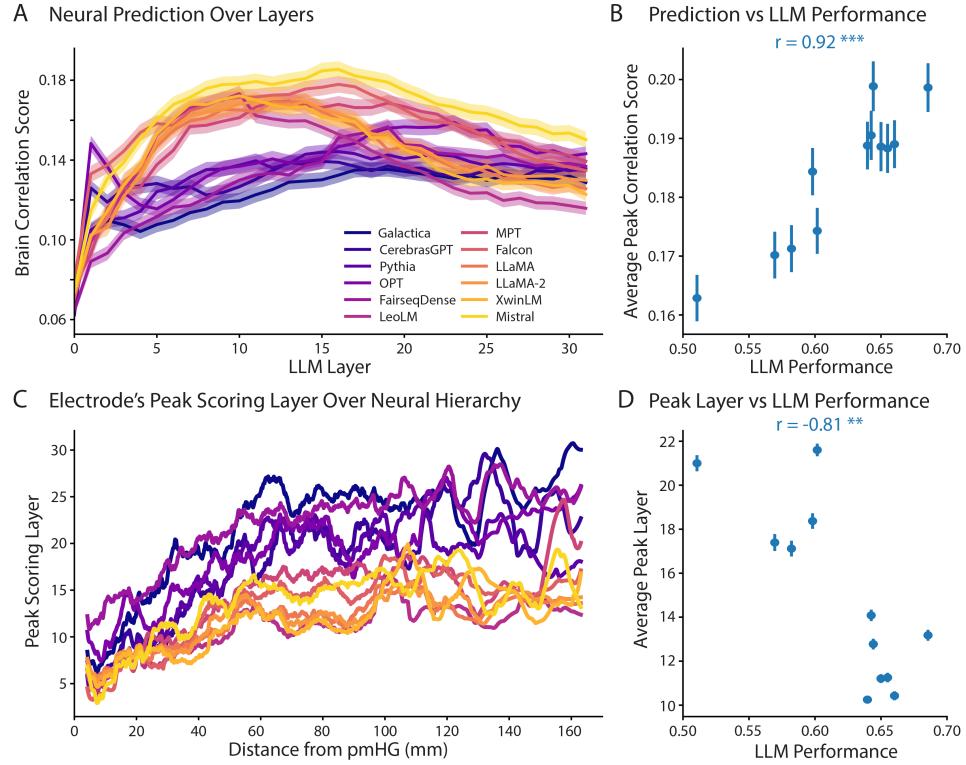


Figure 2: Peak brain correlations and layers relate to LLM performance. A) Average brain correlation over all electrodes for each LLM. LLMs are colored in order of their separately-measured benchmark performance, with blue/purple models performing the worst and yellow models performing the best. Shaded regions indicate standard error of the mean over electrodes. B) The peak correlation over all layers of a given model was computed for each electrode, then averaged over all electrodes. Bars indicate standard error of the mean over electrodes. Average peak correlation score is significantly related to LLM performance (Pearson $r = 0.92, p = 2.24 \times 10^{-5}$). Stars indicate statistical significance level thresholds of $p < 0.05$, $p < 0.01$, and $p < 0.001$ with *, **, and ***, respectively. C) The peak scoring layer of each model was computed for each electrode. Then electrodes were sorted by distance from pmHG and a sliding window average (centered, $n = 50$) was taken across the electrodes of each model to compute the smoothed, local estimate of the most brain-like LLM layer. The peak scoring layer generally increases with distance from pmHG, and the better models (yellow) peak at lower layers compared to the worse models (blue/purple). D) The average peak layer for a given model over all electrodes is shown with bars indicating standard error of the mean. Average peak layer is significantly negatively related to LLM performance (Pearson $r = -0.81, p = 0.0013$).

of hierarchical correspondence between stages of visual cortex processing and image classification networks [28], we sought to compute the alignment between hierarchical feature extraction pathways in brains and models. Although the brain’s exact hierarchical processing stages, analogous to layers of a model, are not perfectly known, we again used the distance from pmHG to quantify the stages of hierarchical processing. We grouped electrodes into bins at 10mm intervals. Then, for each electrode, we normalized the brain similarity scores over layers. Finally, we averaged these layer-wise scores over the electrodes in a bin, producing a set of layer scores which are shown as a single row of the alignment matrix in Fig. 3A. We used the center of mass of this average brain similarity score over layers within each electrode bin to quantify the LLM layer most similar to a given stage of the brain’s hierarchy. Then, we compared the progression of these most-similar LLM layers to the bin distances along the hierarchy, visually finding that some models achieve a more linear increase in LLM layers over bins. We summarize the alignment between the language processing hierarchies of each LLM and the brain using the Pearson correlation between the layer center of mass in each bin and the hierarchical stage of each bin (i.e. the distance of each bin from pmHG) [28]. We

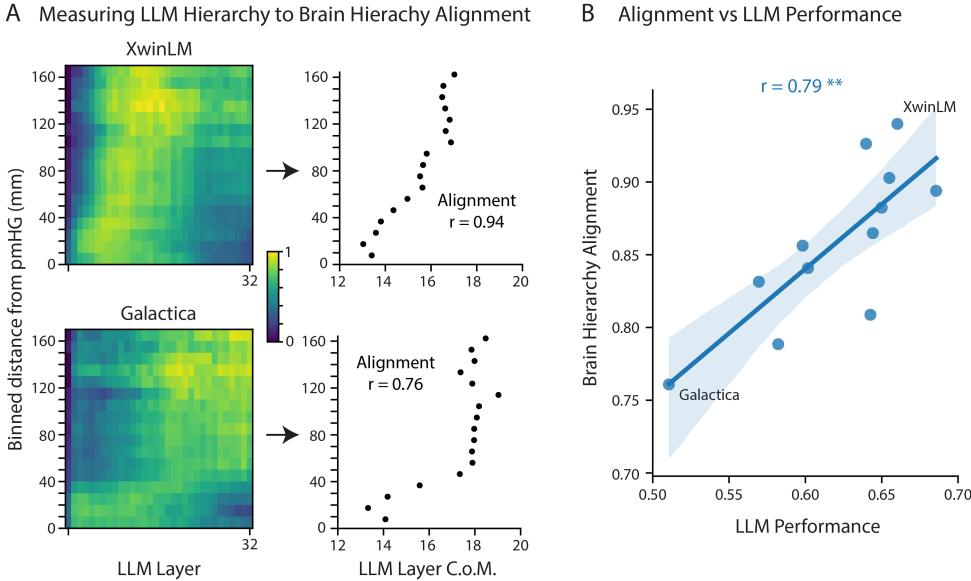


Figure 3: Better LLMs display more brain-like hierarchical processing. A) Examples of computing the brain hierarchy alignment are shown for two models: XwinLM (the model with the highest alignment score) and Galactica (the model with the lowest alignment score). Electrodes were first binned into a hierarchy by distance from pmHG. Within a bin, the correlations over all 32 layers were normalized between 0 and 1 and then averaged over electrodes in the bin, producing one row for each bin in the matrix on the left. The center of mass (C.o.M.) of the distribution of brain similarity scores over LLM layers for each bin was computed and plotted in the scatter plot to the right. The brain hierarchy alignment score was then computed as the Pearson correlation between LLM layer C.o.M. and distance from pmHG. B) A scatter plot of brain hierarchy alignment scores and LLM performance shows a significant positive correlation (Pearson $r = 0.79$, $p = 0.0021$, ** indicates $p < 0.01$). Line and shaded region shows linear regression fit and bootstrapped ($n = 1000$) 95% confidence interval.

illustrate this alignment computation for XwinLM and Galactica, two models which achieve the highest and lowest hierarchy alignment scores, respectively (Fig. 3A). These models also display a stark difference in benchmark performance, with Galactica being the lowest performing LLM. The alignment scores reveal that the better model (XwinLM) exhibits a feature extraction progression more consistent with the brain from early to late-stage processing compared to the bad model. This brain alignment is also highly correlated with LLM performance on the benchmark evaluation tasks (Pearson $r = 0.79$, $p = 0.0021$) (Fig. 3B). We find the same result when using electrode latency to measure the stages of the brain's hierarchical processing, rather than distance from pmHG (Pearson $r = 0.89$, $p = 0.0001$) (Supplementary Fig. 3), which demonstrates that this finding holds for other estimates of the stages of the cortical hierarchy. Additionally, to ensure this effect was not the result of a single subject overpowering the distribution, we separated the even- and odd-numbered subjects and performed the analysis again, finding that brain hierarchy alignment was significantly correlated with LLM performance for each group (Pearson correlation, even subjects: $r = 0.79$, $p = 0.0022$, odd subjects: $r = 0.81$, $p = 0.0013$) (Supplementary Fig. 4). Overall, these findings demonstrate that better-performing LLMs extract features using a hierarchy that more linearly aligns with the brain's hierarchical language processing pathway.

To perform model-to-model comparisons, we used centered kernel alignment (CKA) [29], a method analogous to canonical correlation analysis (CCA) with a nonlinear kernel, which is able to capture similarity between high-dimensional representations like neural network embeddings. We computed the CKA similarity between all pairs of layers for all pairs of models. Thus, each pair of models creates a layer-by-layer similarity matrix describing their embedding similarity. High similarity along the diagonal indicates that the two models extract similar features at the same layers. Higher similarity offset from the diagonal indicates that one model exhibits a delay in extracting similar

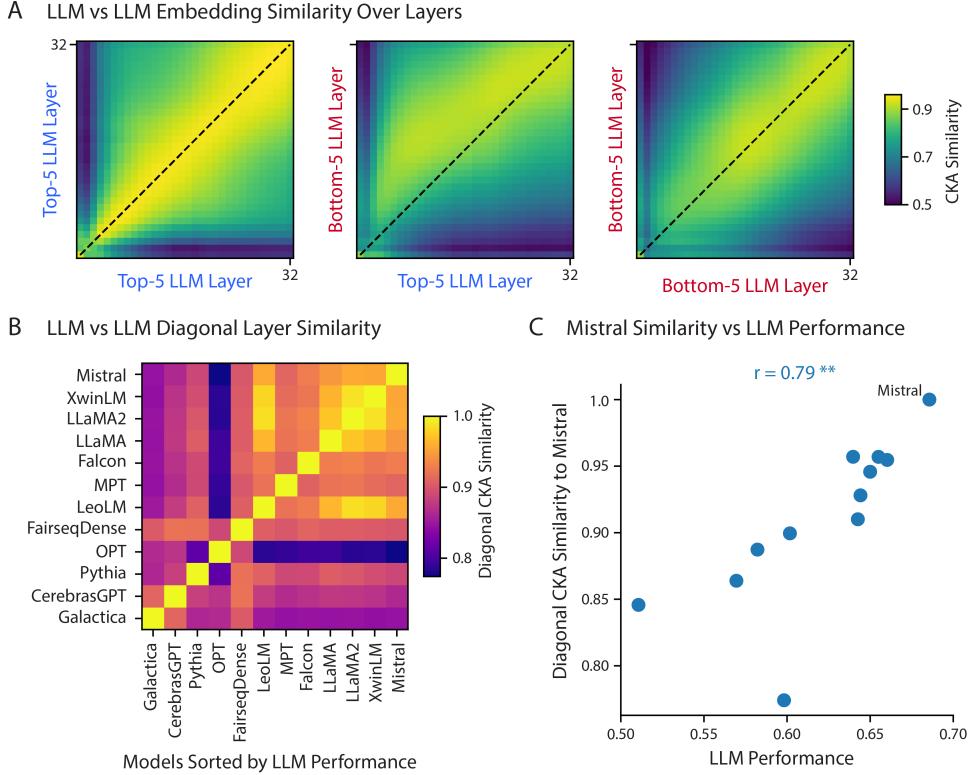


Figure 4: Comparing feature extraction hierarchies between LLMs. A) Layer-by-layer similarity matrices were computed using CKA for every pair of LLMs. LLMs were labeled as either top-5, bottom-5, or excluded, depending on their sorted performance on our LLM benchmark evaluation. Then, similarity matrices between all pairs of top-5 LLMs were averaged and displayed as the “top-5 versus top-5” average similarity matrix in the top left. The same was done to create similarity matrices between the average “top-5 versus bottom-5” as well as the average “bottom-5 versus bottom-5” LLMs. Visually, the “top-5 versus top-5” similarity matrix is highly diagonal, while “bottom-5 versus bottom-5” is less similar in early layers. The “top-5 versus bottom-5” similarity matrix shows an offset diagonal, indicating a delay in feature extraction for the bottom-5 models compared to the top-5. B) Model-by-model diagonal similarity was computed as the average along the diagonal of their similarity matrix. Models are arranged in sorted order of LLM benchmark performance from worst to best. This visually confirms that the best models are fairly similar to each other in layer-wise feature extraction, while worse models are less similar to each other and less similar to the best models. C) The diagonal similarity of each model with Mistral, the best performing LLM, is plotted against the LLM performance, showing a strong positive relationship (Pearson $r = 0.79, p = 0.0022$, ** indicates $p < 0.01$).

features. When grouping these similarity matrices by the top-5 and bottom-5 models based on LLM benchmark performance and averaging within a group, an interesting pattern emerges (Fig. 4A). We find that the top models exhibit a high degree of similarity to each other along the diagonal. On the other hand, the worst models are much less similar to each other in their early layers, and even in their later layers they are less consistent than the top-5-to-top-5 model pairs. Finally, comparing top-5 models to bottom-5 models reveals a striking offset in maximum similarity from the diagonal. This suggests that bad models require more layers to reach a similar level of feature extraction as good models. We summarize the layer-wise feature extraction similarity between each pair of models using the average CKA similarity along the diagonal in their CKA similarity matrix (Fig. 4B). The plot demonstrates that the top-5 models are indeed more similar to each other, with a sub-block of high similarity emerging among the top few models. Since Mistral is the best performing LLM, we look at the diagonal similarity to Mistral of each model and find that a more Mistral-like feature extraction progression correlates strongly with LLM performance (Pearson $r = 0.79, p = 0.0022$)

(Fig. 4C). These results reveal new distinctions between the embeddings of LLMs and suggest that inefficient feature extraction or poor early-layer learning in bad models may contribute to their worse performance and lower brain similarity.

2.3 Contextual Content Supports Brain Hierarchy Alignment

Since the contextual nature of LLM features is critical for their brain similarity compared with non-contextual representations [10, 3, 30], we hypothesized that the amount of contextual information used by a model may also play a key role in determining the alignment between hierarchical feature extraction pathways of LLMs and the brain. We extracted limited-context embeddings from the LLMs by restricting their causal attention mechanism to a certain window of the previous text. Transformer architecture LLMs use tokenizers to separate text into discrete units, so we supplied the models with only the most recent N tokens, sweeping N over a range of values from 1 to 100. A single token input gives the model no context at all. For reference, Mistral’s tokenizer averages 1.15 tokens per word in our stimulus corpus. We then repeated our analysis of model-brain hierarchical alignment (as previously shown in Fig 3B) by computing the correlation between brain hierarchy alignment and LLM performance at each limited context window length. While this correlation is positive for all but the 1-token case, it is only significant for long contextual window lengths of 50 tokens and above (Fig. 5A). This suggests that the brain alignment of LLMs critically depends on the amount of contextual information the model is able to see, which then influences its hierarchical feature extraction mechanism.

Since the correlation between LLM performance and hierarchical alignment is strongly positive for long context lengths, we expected that better-performing models would be better at incorporating contextual information into their language representations. To test this, we quantified the amount of contextual information present in a model’s embeddings by measuring how much its embeddings changed when contextual information was added to the input. We measured the CKA difference ($1 - \text{similarity}_{CKA}(\text{full-context}, \text{1-token})$) of the embeddings of each layer when given the full context compared to the first-layer embeddings when given only a 1-token limited context window. We refer to the average of this CKA difference over all layers as the contextual content of the model’s representations. We find that this contextual content is positively correlated with LLM performance (Spearman $r = 0.66, p = 0.020$) (Fig. 5B). Additionally, it is very strongly correlated with brain similarity (Spearman $r = 0.84, p = 0.0006$) (Fig. 5C). These findings indicate that contextual information plays a crucial role in natural language processing in both natural and artificial language models, and contextual feature extraction enables brain hierarchy alignment in LLMs.

We further investigated the impact of contextual information on neural similarity by computing how much each LLM’s peak similarity score with a given electrode changed when the models were given the full context versus no context (1 token). We then averaged this difference over all LLMs for each electrode and plotted the electrodes on the brain, finding that being given the extra context more greatly improved similarity scores with electrodes in higher-level language processing areas (Fig. 5D). Averaging electrodes within major anatomical regions further quantifies this result, as we find higher average context effects on brain correlation score within the higher-level linguistic-processing area of inferior frontal gyrus (IFG) [31] compared to sensory regions like Heschl’s gyrus (HG) and superior temporal gyrus (STG) (Wilcoxon rank-sum test, $p < 0.05$) (Fig. 5E). Interestingly, the articulatory region of subcentral gyrus, which has also been implicated in high-level linguistic processing [32], displays the highest average score improvement, but due to its high variance it does not meet statistical significance. These results show that contextual information becomes more critical in determining brain similarity further along the spoken language processing hierarchy, which supports previous investigations of high-level linguistic feature encoding in more downstream regions [33, 34]. This finding strengthens the notion that both the brain and LLMs are extracting context along their hierarchies, and that LLMs need contextual information to achieve brain similarity in downstream processing regions. Taken together, our analyses reveal that high-performing LLMs not only extract representations of language that are similar to the brain, but they also use hierarchical feature extraction pathways which more strongly align with that of the brain due to contextual information processing abilities, a finding that uncovers new ways in which the best LLMs are continuously converging toward the brain.

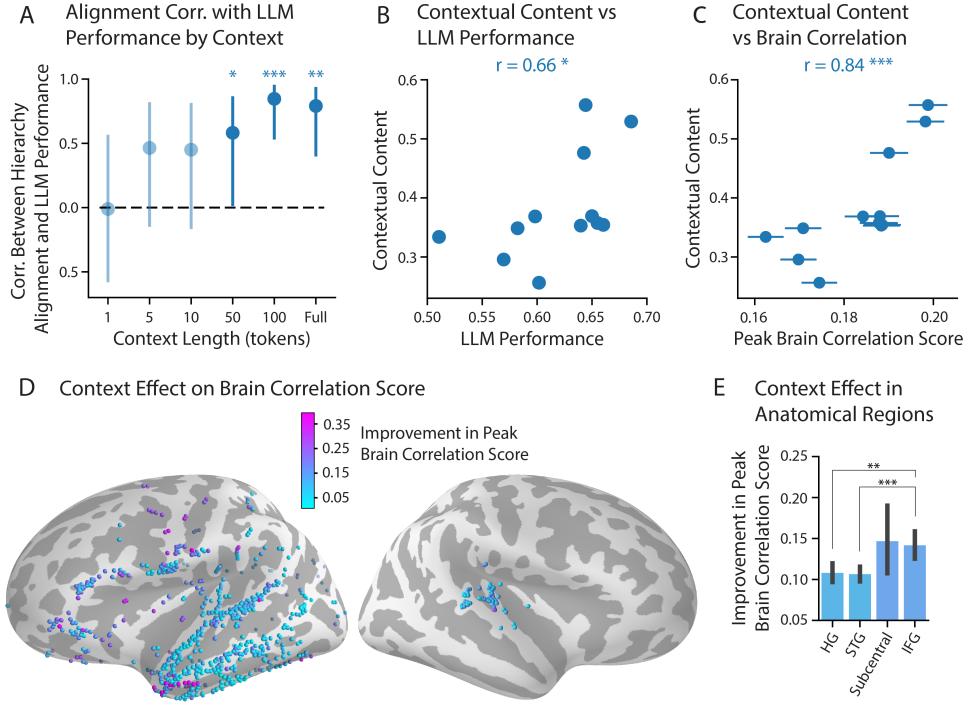


Figure 5: Effect of contextual information. A) Using embeddings from the LLMs when given a certain limited number of the previous tokens as a context window, we performed the analysis of brain hierarchy alignment again. The correlation between LLM performance and brain hierarchy alignment is illustrated by each dot with 95% confidence interval bars, showing only a significant correlation for long contextual windows. Stars illustrate the significance level of the correlation, with *, **, and *** indicating chance levels below 0.05, 0.01, and 0.001, respectively. B) The contextual content of a model’s representations is plotted against its benchmark performance, showing a positive correlation between the two (Spearman $r = 0.66, p = 0.020$). C) Contextual content of each model is plotted against its average peak brain similarity over electrodes, showing a strong correlation (Spearman $r = 0.84, p = 0.0006$). Horizontal lines show standard error of the mean over electrodes for brain similarity. D) Electrodes plotted on the FreeSurfer average inflated brain, colored by the effect of contextual information on peak brain similarity score. E) Bar plot of the average context effect on peak brain similarity score for electrodes within four main anatomical regions along the linguistic hierarchy. Each bar is colored by its value according to the same colormap as used on the brain plot, and error bars show standard error of the mean. Stars indicate significant differences between a pair of regions (Wilcoxon rank-sum test).

3 Discussion

We explored LLMs and their alignment with neural responses during language processing, uncovering several key findings. Firstly, we observed a clear correlation between the language task performance of LLMs and their accuracy in predicting neural responses in the auditory cortex, with higher-performing models exhibiting greater functional alignment with the speech cortex. Secondly, we showed that the models with higher performance on benchmark tasks achieved peak predictive accuracy in earlier layers. In contrast, lower-performing models exhibited a delayed representation, necessitating deeper layers to approach similar levels of brain prediction accuracy. Finally, our study highlights the crucial role of contextual information in both LLMs and brain processing, where the contextual window’s size significantly influenced the difference between better and worse models, with the availability of long-range contextual information driving the high-performing LLMs closer to the brain’s hierarchical pathway. These findings uncover fundamental principles in language processing, highlighting the critical role of hierarchical structure and contextual dependencies in language which give rise to convergent processing strategies in both artificial and biological systems.

3.1 Hierarchical Processing and Inter-Model Comparisons

We found that better-performing LLMs exhibit a more brain-like hierarchy of layers, offering new insights into their language processing. While previous studies have revealed similarities in the hierarchical stages found in the brain and deep neural networks for linguistic [11, 8, 27], acoustic [35, 36], visual [37, 38, 39], and imagined stimuli [40], a distinct approach in our study is the inter-model comparison within a consistent architectural framework. In related work analyzing deep neural networks for vision tasks, recent evidence [28] has shown that better performance can create a less brain-like progression of feature extraction in models when compared to the visual cortex, suggesting that the complex architectures of high-performing image processing networks have steered them away from neural alignment. By examining LLMs based on a single architecture, the stacked transformer decoder [41], we uncover differences in their alignment with the brain’s hierarchical stages during language comprehension. Transformer language models use contextual features to encode linguistic, syntactic, and positional structures [42, 43], and increasingly high-level and context-specific features arise throughout a model’s layers [18, 19]. This may be partly because later layers bind linguistic structures over longer contexts [44]. The crucial observation that such models display brain-like hierarchies resonates with neurobiological findings of hierarchical organization in the auditory and language-related cortex [14, 22, 33, 17, 15, 16, 45, 46]. The convergence of the two systems highlights language’s inherent hierarchical structure as we increasingly form larger units of representation, from articulatory features to phonemes, syllables, words, sentences, and phrases [34, 47, 48]. Our results demonstrate that as LLMs have achieved higher performance, they have done so using feature extraction pathways that more closely resemble the human brain.

3.2 Feature Extraction Efficiency and Contextual Processing

A significant finding of our study is the delayed feature extraction observed in less effective LLMs compared to their higher-performing counterparts. This delay, particularly evident in the early processing stages within transformer models, suggests a slower buildup of relevant linguistic and contextual information [19]. The implications of this observation are multifaceted. Firstly, it challenges the conventional emphasis on the final layers of LLMs [10], instead drawing attention to the critical role of initial layers in efficient language processing [13]. This shift in focus aligns with emerging neuroscience research that underscores the significance of early-stage processing in the human brain for complex cognitive tasks like language processing [46, 34, 48]. Secondly, this delayed representation in less effective models offers insights into potential inefficiencies in their training or design. Given the architectural similarity of models in our study, the variance in feature extraction efficiency among models may reflect differences in training strategies [49] and data quality [50, 51, 20], providing insights for future LLM model development. As LLMs have evolved in recent years, improvements in dataset size and cleanliness as well as architectural changes to increase context length have come along with their performance improvements, and our results show that these improvements have also given rise to greater brain similarity. Furthermore, the observation that higher-performing models utilize early layers more effectively and peak in their brain similarity in middle layers rather than later layers raises intriguing questions about the role of subsequent layers. It is possible that these later layers are engaged in next-level contextual integration and feature extraction, potentially analogous to higher-order stimulus integration to support cognitive functions in the human brain [52, 53]. Alternatively, this finding could point to a limitation in our current methodologies, such as limited iEEG coverage, the simplicity of the speech comprehension task, or the fact that LLMs are not explicitly trained to perform comprehension, but rather next-word prediction, which is slightly different from the speech listening comprehension task the subjects performed. Our iEEG recordings include broad coverage of speech processing regions, especially acoustic sensory regions like HG and STG, which, although critical for spoken language processing, represent a slightly different aspect of linguistic feature extraction than the token-level processing that transformer architecture LLMs begin with. Answering these questions is crucial for enriching our understanding of artificial language processing.

The influence of contextual information on brain similarity and LLM benchmark scores also points to specific avenues that may improve model performance on language tasks. Ensuring that models are able to extract long context windows, such as by using architectures that allow for long context windows [54] and utilizing training data that is rich in long context information, could enhance LLM performance further beyond simply scaling up a model’s parameter size. Transformer-based LLMs have been shown to suffer from unequal contextual information extraction when the prior

context occurs at different distances from the target [55], supporting the notion that improving the robustness of modern LLMs to varying context lengths may lead to performance improvements. Our investigation offers a unique lens through which to view the parallels and divergences between machine learning and human cognitive development.

3.3 Convergence to Brain-Like Models for Human-Level Artificial General Intelligence

The convergence of LLMs and human speech processing may suggest that certain fundamental principles underlying efficient language processing might be common to both artificial and biological systems. The human brain’s language capabilities have developed as an adaptive response to complex communication needs, optimizing for efficiency and versatility [56]. Our findings suggest that LLM architectures and processing strategies are gravitating towards these same principles, mimicking the brain’s evolutionary adaptations for language. LLMs are trained without consideration for brain similarity, yet they have become increasingly brain-like in their feature extraction and hierarchical processing. Brain-like processing may represent an optimal solution to language modeling found by evolution [57], although subject to biological constraints, and our results suggest that modern LLM training focused on performance optimization may have placed these models on a similar path. In our study, Mistral, the top-performing model, stands as a prime example of this convergence, where the degree of similarity of a model’s embeddings to those of Mistral is highly correlated with performance and brain similarity. This evolution towards an optimal brain-like model offers an intriguing suggestion regarding artificial general intelligence (AGI). While not clearly defined, AGI can be quantified as human-level performance on a broad set of benchmarks [58]. Our findings suggest that developing models mimicking human neural processing strategies [59], rather than solely focusing on augmenting computational power or diversifying learning algorithms [60], could accelerate the development of models that behave on par with human performance. Hence, brain similarity could be a useful evaluation and optimization metric for future model development.

Our research marks a significant stride in understanding the parallels between large language models and human brain processes in language comprehension, by revealing the intricate relationship between internal model representation, model performance, and neural predictive accuracy. Our findings enhance the understanding of LLMs and offer new insights into the cognitive mechanisms underlying human language processing.

4 Methods

4.1 Human Intracranial Recordings

Eight subjects undergoing clinical evaluation for drug-resistant epilepsy participated in the study. Electrodes were implanted intracranially (iEEG) with the clinical goal of identifying epileptogenic foci for surgical removal. Any electrodes showing signs of epileptiform discharges, as identified by an epileptologist, were not analyzed in this study. Prior to electrode implantation, all subjects provided written informed consent for research participation. The research protocol was approved by the institutional review board at North Shore University Hospital.

Subjects listened to naturalistic recordings of voice actors reading passages from stories and conversations. To ensure the subjects were paying attention to the stimuli, one of the voices in the recording occasionally directed a question at the listener directly, or the stories were paused and the subject was asked a question, to check their understanding. The subjects were able to effectively answer each question. These pauses separated the stimulus into separate passages.

The envelope of the high-gamma band (70-150 Hz) of the raw neural recordings was computed using the Hilbert transform [61] and downsampled to 100 Hz. This signal was used as the neural response due to its correlation with neuronal firing rates [62, 63] and its common use in auditory neuroscience research [64, 65]. We restricted our analysis to speech-responsive electrodes, which we estimated using a t-test between each electrode’s response to the first second of the stimulus compared to last second of silence preceding it (FDR corrected, $p < 0.05$ [21]), which left 707 electrodes for analysis. We extracted average word responses from each electrode by taking the average high-gamma signal value in a 100ms window around the midpoint of each word.

4.2 Large Language Models

We analyzed 12 LLMs of approximately 7 billion parameters downloaded from Hugging Face and implemented with its `Transformers` library [66], including the most recent and most popular open-source LLMs. We selected these models by searching the Hugging Face Hub for 7 billion parameter models, then using as many of the trending or most-downloaded models that we were able to run without issue.

We computed two similar evaluation metrics to those used by LLaMA 2 [20]: Reading Comprehension and Commonsense Reasoning. As measures of English language understanding, these are both highly related to the listening comprehension task which was performed by the human subjects in the study. As in [20], these metrics were created by averaging the model’s performance on a certain set of related tasks. All individual benchmarks were computed for each model using the Language Model Evaluation Harness [67] on Github.

- Reading Comprehension - This metric was the average 0-shot performance of a model on SQuAD 2.0 [68] and BoolQ [69].
- Commonsense Reasoning - This metric consists of the average 0-shot performance on OpenBookQA [70], PIQA [71], HellaSwag [72], and Winogrande [73].

Overall LLM Performance was computed as the average Reading Comprehension and Commonsense Reasoning scores.

The models used, and their benchmark performance and overall LLM performance scores, are shown in Table 1.

Models Used	Reading Comprehension	Commonsense Reasoning	LLM Performance
Galactica-6.7B [74]	0.486	0.535	0.511
CerebrasGPT-6.7B [75]	0.565	0.575	0.570
Pythia-6.9B [76]	0.568	0.597	0.582
OPT-6.7B [77]	0.581	0.616	0.598
FairseqDense-6.7B [78]	0.575	0.628	0.602
LeoLM-7B [79]	0.634	0.646	0.640
MPT-7B [80]	0.620	0.665	0.643
Falcon-7B [81]	0.619	0.669	0.644
LLaMA-7B [82]	0.626	0.674	0.650
LLaMA2-7B [20]	0.639	0.671	0.655
XwinLM-7B [83]	0.648	0.673	0.660
Mistral-7B [84]	0.669	0.703	0.686

Table 1: All models used in the study, along with their computed benchmark performances.

In order to extract LLM embeddings for each stimulus passage (approximately 30-60 seconds when spoken), we fed the text to the model and extracted the embeddings of each layer when given a causal attention mask. When limiting the contextual window of the model, the attention mask was truncated to only include the most recent N tokens. For multi-token words, we used the embedding of the last token in the word. Thus, for each passage, we extracted a tensor of embeddings of shape $(L_{\text{layers}}, N_{\text{words}}, D_{\text{dimensions}})$ from each model.

4.3 Ridge Regression Mapping from Embeddings to Neural Responses

We performed PCA to reduce the dimensionality of each model’s embeddings to 500 components. For a given model, PCA was performed for each layer separately. Then, we fit 10-fold cross-validated ridge regression models to predict the average word responses from each layer’s embeddings, sweeping over a range of regularization parameters for each training fold, using `scikit-learn`’s `RidgeCV` model [85].

4.4 Electrode Localization and Brain Plotting

Each subject’s electrode positions were mapped to the subject’s brain using iELVis [86] to perform co-registration between pre- and post-implant MRI scans. Then, the subject-specific electrode locations were mapped to the FreeSurfer average brain [87]. Euclidean distance from posteromedial HG (TE1.1) [23] was computed in this average brain, since TE1.1 is a landmark of primary auditory cortex [24, 45, 25, 26]. When visualizing electrodes on the average brain, all subdural electrodes were snapped to the nearest surface point.

4.5 Comparing LLMs with Centered Kernel Alignment

To estimate the similarity between high-dimensional embeddings of different models, we used CKA [29], a similarity metric which is related to CCA but has been shown to perform well in high-dimensional scenarios between neural network features. We used the RBF kernel to allow for nonlinear similarity measurement. For a given pair of models, we computed the CKA similarity between the embeddings of one layer of the first model with another layer of the second model. Iterating over all pairs of layers for those two models produced a single similarity matrix. These similarity matrices were then grouped by whether they described a comparison between two models in the top-5 of all LLMs for benchmark performance, one model in the top-5 and the other in the bottom-5, or two models in the bottom-5, and then averaged.

4.6 Data and Code Availability

Although the iEEG recordings used in this study cannot be made publicly available, they can be requested from the author [N.M.]. Code for preprocessing neural recordings, including extracting the high-gamma envelope and identifying responsive electrodes is available in the naplib-python package [88].

Acknowledgement

This work was funded by the National Institutes of Health, the National Institute on Deafness and Other Communication Disorders, and the National Science Foundation Graduate Research Fellowship Program. The funders had no role in study design, data collection and analysis, decision to publish or preparation of the manuscript.

References

- [1] Mariya Toneva and Leila Wehbe. Interpreting and improving natural-language processing (in machines) with natural language-processing (in the brain). *Advances in neural information processing systems*, 32, 2019.
- [2] Samira Abnar, Lisa Beinborn, Rochelle Choenni, and Willem Zuidema. Blackbox meets blackbox: Representational similarity and stability analysis of neural language models and brains. *arXiv preprint arXiv:1906.01539*, 2019.
- [3] Martin Schrimpf, Idan Asher Blank, Greta Tuckute, Carina Kauf, Eghbal A Hosseini, Nancy Kanwisher, Joshua B Tenenbaum, and Evelina Fedorenko. The neural architecture of language: Integrative modeling converges on predictive processing. *Proceedings of the National Academy of Sciences*, 118(45):e2105646118, 2021.
- [4] Charlotte Caucheteux and Jean-Rémi King. Language processing in brains and deep neural networks: computational convergence and its limits. *BioRxiv*, pages 2020–07, 2020.
- [5] Eghbal A Hosseini, Martin Schrimpf, Yian Zhang, Samuel Bowman, Noga Zaslavsky, and Evelina Fedorenko. Artificial neural network language models align neurally and behaviorally with humans even after a developmentally realistic amount of training. *BioRxiv*, pages 2022–10, 2022.
- [6] Andrew James Anderson, Douwe Kiela, Jeffrey R Binder, Leonardo Fernandino, Colin J Humphries, Lisa L Conant, Rajeev DS Raizada, Scott Grimm, and Edmund C Lalor. Deep artificial neural networks reveal a distributed cortical network encoding propositional sentence-level meaning. *Journal of Neuroscience*, 41(18):4100–4119, 2021.

- [7] Charlotte Caucheteux, Alexandre Gramfort, and Jean-Remi King. Disentangling syntax and semantics in the brain with deep networks. In *International conference on machine learning*, pages 1336–1348. PMLR, 2021.
- [8] Charlotte Caucheteux and Jean-Rémi King. Brains and algorithms partially converge in natural language processing. *Communications biology*, 5(1):134, 2022.
- [9] Jingyuan Sun, Shaonan Wang, Jiajun Zhang, and Chengqing Zong. Neural encoding and decoding with distributed sentence representations. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2):589–603, 2020.
- [10] Ariel Goldstein, Zaid Zada, Eliav Buchnik, Mariano Schain, Amy Price, Bobbi Aubrey, Samuel A Nastase, Amir Feder, Dotan Emanuel, Alon Cohen, et al. Shared computational principles for language processing in humans and deep language models. *Nature neuroscience*, 25(3):369–380, 2022.
- [11] Charlotte Caucheteux, Alexandre Gramfort, and Jean-Rémi King. Evidence of a predictive coding hierarchy in the human brain listening to speech. *Nature human behaviour*, 7(3):430–441, 2023.
- [12] Richard Antonello, Aditya Vaidya, and Alexander G Huth. Scaling laws for language encoding models in fmri. *arXiv preprint arXiv:2305.11863*, 2023.
- [13] Richard Antonello and Alexander Huth. Predictive coding or just feature discovery? an alternative account of why language models fit brain data. *Neurobiology of Language*, pages 1–16, 2023.
- [14] Gregory Hickok and David Poeppel. The cortical organization of speech processing. *Nature reviews neuroscience*, 8(5):393–402, 2007.
- [15] Uri Hasson, Eunice Yang, Ignacio Vallines, David J Heeger, and Nava Rubin. A hierarchy of temporal receptive windows in human cortex. *Journal of Neuroscience*, 28(10):2539–2550, 2008.
- [16] Yulia Lerner, Christopher J Honey, Lauren J Silbert, and Uri Hasson. Topographic mapping of a hierarchy of temporal receptive windows using a narrated story. *Journal of Neuroscience*, 31(8):2906–2915, 2011.
- [17] Nai Ding, Lucia Melloni, Aotian Yang, Yu Wang, Wen Zhang, and David Poeppel. Characterizing neural entrainment to hierarchical linguistic units using electroencephalography (eeg). *Frontiers in human neuroscience*, 11:481, 2017.
- [18] Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. *arXiv preprint arXiv:1909.00512*, 2019.
- [19] Ian Tenney, Dipanjan Das, and Ellie Pavlick. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*, 2019.
- [20] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.
- [21] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics*, pages 65–70, 1979.
- [22] Tatyana O Sharpee, Craig A Atencio, and Christoph E Schreiner. Hierarchical representations in the auditory cortex. *Current opinion in neurobiology*, 21(5):761–767, 2011.
- [23] Patricia Morosan, Jorg Rademacher, Axel Schleicher, Katrin Amunts, Thorsten Schormann, and Karl Zilles. Human primary auditory cortex: cytoarchitectonic subdivisions and mapping into a spatial reference system. *Neuroimage*, 13(4):684–701, 2001.
- [24] Simon Baumann, Christopher I Petkov, and Timothy D Griffiths. A unified framework for the organization of the primate auditory cortex. *Frontiers in systems neuroscience*, 7:11, 2013.
- [25] Sam V Norman-Haignere and Josh H McDermott. Neural responses to natural and model-matched stimuli reveal distinct computations in primary and nonprimary auditory cortex. *PLoS biology*, 16(12):e2005127, 2018.
- [26] Gavin Mischler, Menoua Keshishian, Stephan Bickel, Ashesh D Mehta, and Nima Mesgarani. Deep neural networks effectively model neural adaptation to changing background noise and suggest nonlinear noise filtering methods in auditory cortex. *NeuroImage*, 266:119819, 2023.

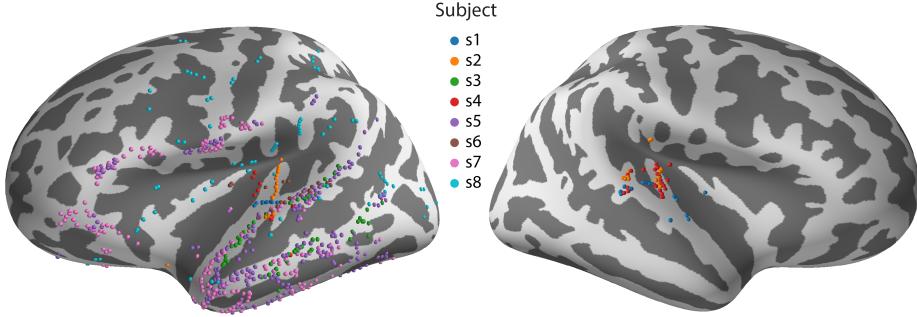
- [27] Sreejan Kumar, Theodore R Sumers, Takateru Yamakoshi, Ariel Goldstein, Uri Hasson, Kenneth A Norman, Thomas L Griffiths, Robert D Hawkins, and Samuel A Nastase. Reconstructing the cascade of language processing in the brain using the internal computations of a transformer-based language model. *BioRxiv*, pages 2022–06, 2022.
- [28] Soma Nonaka, Kei Majima, Shuntaro C Aoki, and Yukiyasu Kamitani. Brain hierarchy score: Which deep neural networks are hierarchically brain-like? *IScience*, 24(9), 2021.
- [29] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR, 2019.
- [30] Charlotte Caucheteux, Alexandre Gramfort, and Jean-Rémi King. Deep language algorithms predict semantic comprehension from brain activity. *Scientific reports*, 12(1):16327, 2022.
- [31] Sergi G Costafreda, Cynthia HY Fu, Lucy Lee, Brian Everitt, Michael J Brammer, and Anthony S David. A systematic review and quantitative appraisal of fmri studies of verbal fluency: role of the left inferior frontal gyrus. *Human brain mapping*, 27(10):799–810, 2006.
- [32] Sophie Arana, André Marquand, Annika Hultén, Peter Hagoort, and Jan-Mathijs Schoffelen. Sensory modality-independent activation of the brain network for language. *Journal of neuroscience*, 40(14):2914–2924, 2020.
- [33] Jingwei Sheng, Li Zheng, Bingjiang Lyu, Zhehang Cen, Lang Qin, Li Hai Tan, Ming-Xiong Huang, Nai Ding, and Jia-Hong Gao. The cortical maps of hierarchical linguistic structures during speech perception. *Cerebral cortex*, 29(8):3232–3240, 2019.
- [34] Menoua Keshishian, Serdar Akkol, Jose Herrero, Stephan Bickel, Ashesh D Mehta, and Nima Mesgarani. Joint, distributed and hierarchically organized encoding of linguistic features in the human auditory cortex. *Nature Human Behaviour*, 7(5):740–753, 2023.
- [35] Bruno L Giordano, Michele Esposito, Giancarlo Valente, and Elia Formisano. Intermediate acoustic-to-semantic representations link behavioral and neural responses to natural sounds. *Nature Neuroscience*, 26(4):664–672, 2023.
- [36] Greta Tuckute, Jenelle Feather, Dana Boebinger, and Josh H McDermott. Many but not all deep neural network audio models capture brain responses and exhibit correspondence between model stages and brain regions. *Plos Biology*, 21(12):e3002366, 2023.
- [37] Nikolaus Kriegeskorte. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual review of vision science*, 1:417–446, 2015.
- [38] Radoslaw Martin Cichy, Aditya Khosla, Dimitrios Pantazis, Antonio Torralba, and Aude Oliva. Comparison of deep neural networks to spatio-temporal cortical dynamics of human visual object recognition reveals hierarchical correspondence. *Scientific reports*, 6(1):27755, 2016.
- [39] Nicholas J Sexton and Bradley C Love. Reassessing hierarchical correspondences between brain and deep networks through direct interface. *Science Advances*, 8(28):eabm2219, 2022.
- [40] Tomoyasu Horikawa and Yukiyasu Kamitani. Hierarchical neural representation of dreamed objects revealed by brain decoding with deep neural network features. *Frontiers in computational neuroscience*, 11:4, 2017.
- [41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [42] Joe O’Connor and Jacob Andreas. What context features can transformer language models use? *arXiv preprint arXiv:2106.08367*, 2021.
- [43] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. What does bert look at? an analysis of bert’s attention. *arXiv preprint arXiv:1906.04341*, 2019.
- [44] David Skrill and Samuel Victor Norman-Haignere. Large language models transition from integrating across position-yoked, exponential windows to structure-yoked, power-law windows. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.
- [45] Sam V Norman-Haignere, Laura K Long, Orrin Devinsky, Werner Doyle, Ifeoma Irobunda, Edward M Merricks, Neil A Feldstein, Guy M McKhann, Catherine A Schevon, Adeen Flinker, et al. Multiscale temporal integration organizes hierarchical computation in human auditory cortex. *Nature human behaviour*, 6(3):455–469, 2022.

- [46] Wendy A de Heer, Alexander G Huth, Thomas L Griffiths, Jack L Gallant, and Frédéric E Theunissen. The hierarchical cortical organization of human speech processing. *Journal of Neuroscience*, 37(27):6539–6557, 2017.
- [47] Giovanni M Di Liberto, Jingping Nie, Jeremy Yeaton, Bahar Khalighinejad, Shihab A Shamma, and Nima Mesgarani. Neural representation of linguistic feature hierarchy reflects second-language proficiency. *Neuroimage*, 227:117586, 2021.
- [48] Xue L Gong, Alexander G Huth, Fatma Deniz, Keith Johnson, Jack L Gallant, and Frédéric E Theunissen. Phonemic segmentation of narrative speech in human cerebral cortex. *Nature communications*, 14(1):4309, 2023.
- [49] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*, 2023.
- [50] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.
- [51] Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. *arXiv preprint arXiv:2107.06499*, 2021.
- [52] Alexander G Huth, Wendy A De Heer, Thomas L Griffiths, Frédéric E Theunissen, and Jack L Gallant. Natural speech reveals the semantic maps that tile human cerebral cortex. *Nature*, 532(7600):453–458, 2016.
- [53] Elliot Murphy, Kiefer J Forseth, Cristian Donos, Kathryn M Snyder, Patrick S Rollo, and Nitin Tandon. The spatiotemporal dynamics of semantic integration in the human brain. *Nature Communications*, 14(1):6336, 2023.
- [54] Wenhan Xiong, Jingyu Liu, Igor Molybog, Hejia Zhang, Prajjwal Bhargava, Rui Hou, Louis Martin, Rashi Rungta, Karthik Abinav Sankararaman, Barlas Oguz, et al. Effective long-context scaling of foundation models. *arXiv preprint arXiv:2309.16039*, 2023.
- [55] Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *arXiv preprint arXiv:2307.03172*, 2023.
- [56] Steven Pinker and Paul Bloom. Natural language and natural selection. *Behavioral and brain sciences*, 13(4):707–727, 1990.
- [57] Terrence William Deacon. *The symbolic species: The co-evolution of language and the brain*. Number 202. WW Norton & Company, 1997.
- [58] Ben Goertzel. Artificial general intelligence: concept, state of the art, and future prospects. *Journal of Artificial General Intelligence*, 5(1):1, 2014.
- [59] Lin Zhao, Lu Zhang, Zihao Wu, Yuzhong Chen, Haixing Dai, Xiaowei Yu, Zhengliang Liu, Tuo Zhang, Xintao Hu, Xi Jiang, et al. When brain-inspired ai meets agi. *Meta-Radiology*, page 100005, 2023.
- [60] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.
- [61] Erik Edwards, Maryam Soltani, Won Kim, Sarang S Dalal, Srikanth S Nagarajan, Mitchel S Berger, and Robert T Knight. Comparison of time-frequency responses and the event-related potential to auditory speech stimuli in human cortex. *Journal of neurophysiology*, 102(1):377–386, 2009.
- [62] Supratim Ray and John HR Maunsell. Different origins of gamma rhythm and high-gamma activity in macaque visual cortex. *PLoS biology*, 9(4):e1000610, 2011.
- [63] Mitchell Steinschneider, Yonatan I Fishman, and Joseph C Arezzo. Spectrotemporal analysis of evoked and induced electroencephalographic responses in primary auditory cortex (a1) of the awake monkey. *Cerebral Cortex*, 18(3):610–625, 2008.
- [64] Nima Mesgarani, Connie Cheung, Keith Johnson, and Edward F Chang. Phonetic feature encoding in human superior temporal gyrus. *Science*, 343(6174):1006–1010, 2014.

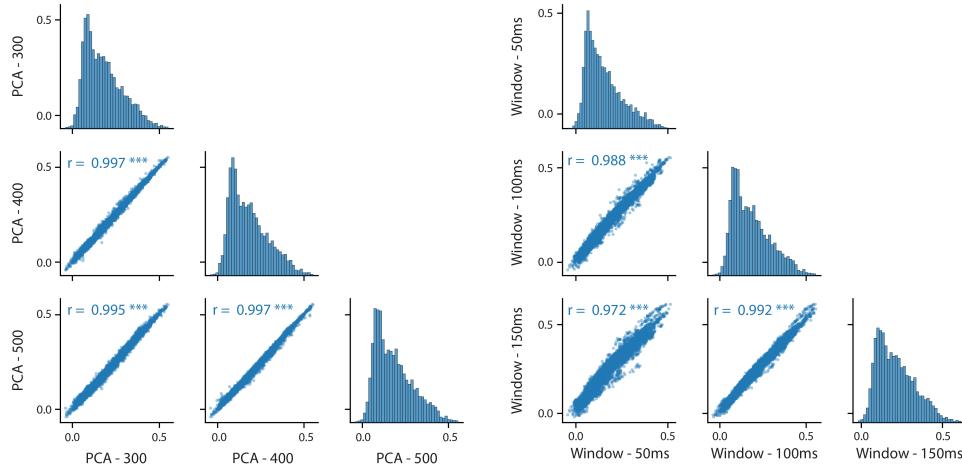
- [65] Kristofer E Bouchard, Nima Mesgarani, Keith Johnson, and Edward F Chang. Functional organization of human sensorimotor cortex for speech articulation. *Nature*, 495(7441):327–332, 2013.
- [66] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierrick Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface’s transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.
- [67] Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, September 2021.
- [68] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don’t know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*, 2018.
- [69] Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*, 2019.
- [70] Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*, 2018.
- [71] Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439, 2020.
- [72] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.
- [73] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.
- [74] Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*, 2022.
- [75] Nolan Dey, Gurpreet Gosal, Hemant Khachane, William Marshall, Ribhu Pathria, Marvin Tom, Joel Hestness, et al. Cerebras-gpt: Open compute-optimal language models trained on the cerebras wafer-scale cluster. *arXiv preprint arXiv:2304.03208*, 2023.
- [76] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O’Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR, 2023.
- [77] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuhui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.
- [78] Mikel Artetxe, Shruti Bhosale, Naman Goyal, Todor Mihaylov, Myle Ott, Sam Shleifer, Xi Victoria Lin, Jingfei Du, Srinivasan Iyer, Ramakanth Pasunuru, et al. Efficient large scale language modeling with mixtures of experts. *arXiv preprint arXiv:2112.10684*, 2021.
- [79] Laion leolm: Linguistically enhanced open language model. <https://huggingface.co/LeoLM/leo-hessianai-13b>. Accessed: 2023-10-01.
- [80] MosaicML NLP Team. Introducing mpt-7b: A new standard for open-source, commercially usable llms. <https://www.mosaicml.com/blog/mpt-7b>, 2023.
- [81] Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, et al. The falcon series of open language models. *arXiv preprint arXiv:2311.16867*, 2023.
- [82] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

- [83] Xwin-lm: Powerful, stable, and reproducible llm alignment. <https://huggingface.co/Xwin-LM/Xwin-LM-7B-V0.2>. Accessed: 2023-10-01.
- [84] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.
- [85] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.
- [86] David M Groppe, Stephan Bickel, Andrew R Dykstra, Xiuyuan Wang, Pierre Mégevand, Manuel R Mercier, Fred A Lado, Ashesh D Mehta, and Christopher J Honey. ielvis: An open source matlab toolbox for localizing and visualizing human intracranial electrode data. *Journal of neuroscience methods*, 281:40–48, 2017.
- [87] Bruce Fischl, André Van Der Kouwe, Christophe Destrieux, Eric Halgren, Florent Ségonne, David H Salat, Evelina Busa, Larry J Seidman, Jill Goldstein, David Kennedy, et al. Automatically parcellating the human cerebral cortex. *Cerebral cortex*, 14(1):11–22, 2004.
- [88] Gavin Mischler, Vinay Raghavan, Menoua Keshishian, and Nima Mesgarani. naplib-python: Neural acoustic data processing and analysis tools in python. *Software Impacts*, 17:100541, 2023.

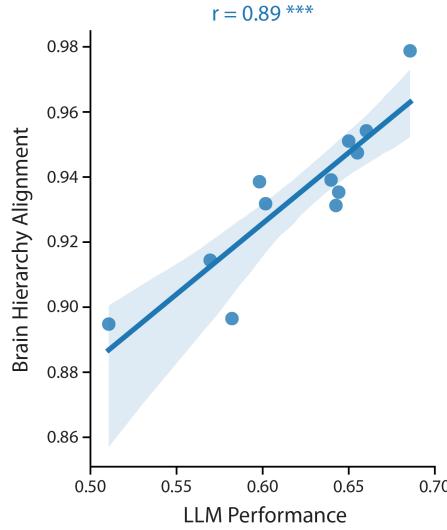
5 Supplementary Figures



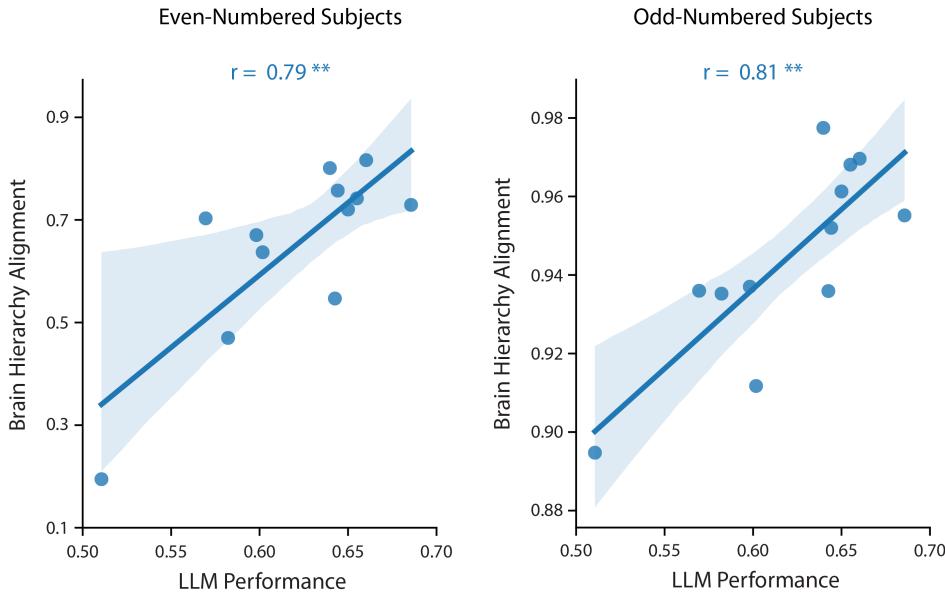
Supplementary Figure 1: Subject-wise electrode localization. Electrodes are plotted on the inflated Freesurfer average brain and are colored by their corresponding subject identity.



Supplementary Figure 2: Effect of regression hyperparameters on scores. The left plot shows the pairwise effects on the peak brain similarity scores when altering the number of principal components of the LLM embeddings used for computing scores with ridge regression, keeping a 100ms window size constant. The right plot shows the pairwise effects of altering the width of the averaging window around word centers for estimating neural responses to words, keeping the PCA dimensionality of 500 constant. Along each plot's diagonal is the marginal distribution for that hyperparameter setting. The off-diagonal plots display scatter plots of all the peak-scores for all models together for one hyperparameter setting against another. Each dot represents the peak brain correlation score for one model-electrode pair. All pairs of settings produce scores which are highly correlated, as written in each subplot (Pearson correlation, *** indicates $p < 0.001$).



Supplementary Figure 3: Hierarchy alignment by model when using electrode lag instead of distance to estimate neural hierarchy. We used the electrode lag, instead of distance from primary auditory cortex, to bin electrodes into a hierarchy with a bin-width of 40ms. We estimated electrode lag using the peak of a 1D temporal receptive field fitted for each electrode to predict its response from the acoustic envelope of the stimulus sound. We then performed the same analysis as shown in Fig. 3, reproducing Fig. 3B with new brain hierarchy alignment for each model. These alignment values are similarly significantly correlated with LLM performance (Pearson $r = 0.89, p = 0.0001$).



Supplementary Figure 4: Hierarchy alignment patterns hold for partial subject groupings. Splitting the electrodes based on whether they came from even- or odd-numbered subjects, we performed the same analyses as in Fig. 3B. Both subject groups show that brain hierarchy alignment is significantly correlated with LLM performance (Pearson correlations in figure, even $p = 0.0022$, odd $p = 0.0013$) demonstrating that this effect is not the result of a single outlier subject.

EEG-GPT: EXPLORING CAPABILITIES OF LARGE LANGUAGE MODELS FOR EEG CLASSIFICATION AND INTERPRETATION

A PREPRINT

✉ **Jonathan W. Kim, BS**
Carle Illinois College of Medicine
Urbana-Champaign, IL
jwk7@illinois.edu

✉ **Ahmed Alaa, PhD**
Department of EECS
UC Berkeley
Berkeley, CA

✉ **Danilo Bernardo, MD**
Department of Neurology
UC San Francisco
San Francisco, CA
dbernardoj@gmail.com

February 6, 2024

ABSTRACT

Rationale In conventional machine learning (ML) approaches applied to electroencephalography (EEG), this is often a limited focus, isolating specific brain activities occurring across disparate temporal scales (from transient spikes in milliseconds to seizures lasting minutes) and spatial scales (from localized high-frequency oscillations to global sleep activity). This siloed approach limits the development EEG ML models that exhibit multi-scale electrophysiological understanding and classification capabilities. Moreover, typical ML EEG approaches utilize black-box approaches, limiting their interpretability and trustworthiness in clinical contexts. Thus, we propose EEG-GPT, a unifying approach to EEG classification that leverages advances in large language models (LLM). EEG-GPT achieves excellent performance comparable to current state-of-the-art deep learning methods in classifying normal from abnormal EEG in a few-shot learning paradigm utilizing only 2% of training data. Furthermore, it offers the distinct advantages of providing intermediate reasoning steps and coordinating specialist EEG tools across multiple scales in its operation, offering transparent and interpretable step-by-step verification, thereby promoting trustworthiness in clinical contexts.

Keywords Large language models · EEG · machine learning

1 Introduction

Large language models (LLMs) such as ChatGPT have garnered substantial attention in the media and among the machine learning (ML) community. LLMs represent a pivotal paradigm shift in artificial intelligence (AI), consisting of transformer architectures substantially larger in scale compared to their predecessors, such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks [1], and leverage internet-scale text corpora, thus excelling not only on text completion tasks but demonstrating emergent capabilities in rudimentary language reasoning [2, 3].

LLMs display several features conducive to the small data regime present in most EEG datasets, where the largest datasets typically have on the order of only thousands of EEGs. Primarily, LLMs have the capability to perform few- and even zero-shot learning [4]. Recent research has investigated how LLMs can perform few-shot learning in domains ranging from cancer drug synergy prediction to cardiac signal analysis [5, 6]. Other work has demonstrated the ability of LLMs to outperform experts in annotating political Twitter messages with zero-shot learning [7]. Additionally, previous work has shown that transformer architectures are capable of utilizing in-context learning for zero-shot tasks – in other words, utilizing information provided in the prompt in order to yield better performance on various tasks [8].

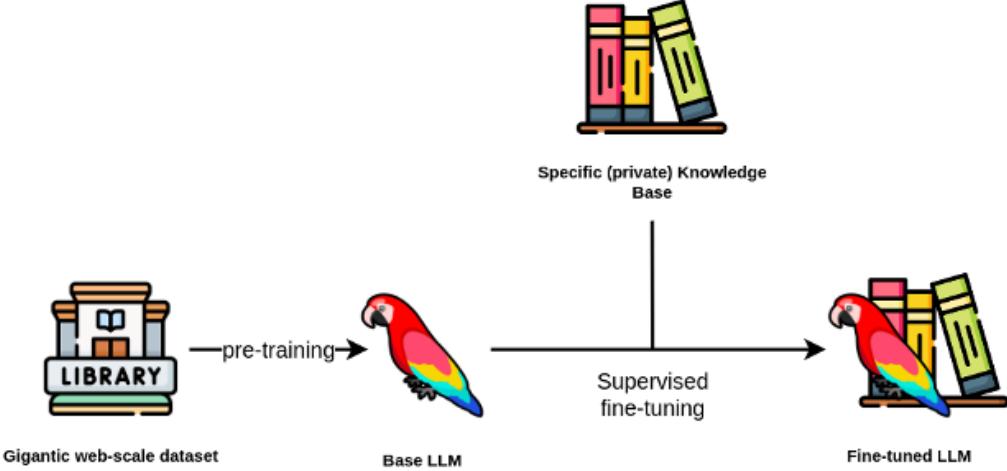


Figure 1: Diagram depicting the process of fine-tuning large language models [9]

LLMs also appear to have the ability to plan and carry out intermediate reasoning steps when asked to solve complex problems. Lightman et al. investigate this property by training an LLM to solve complex competition-level math problems in a stepwise manner that is transparent and human-interpretable [2]. Previous work has investigated using prompting strategies such as Chain of Thought [3], which have been shown to improve the ability of LLMs to perform multi-step computations. More recently, Yao et al. developed a framework called Tree of Thoughts, an extension of Chain of Thought approaches which constructs a decision tree to be explored by an LLM in order to facilitate solving complex problems requiring strategic lookahead and backtracking [10].

Finally, LLMs appear to be capable of synergizing inputs from other computational “experts” as a human might. In other words, when asked to solve a complex problem, if given access to other specialized computational tools, LLMs exhibit the capability to coordinate passing appropriate inputs to those specialist tools then synergizing their outputs to come to a result. Yang et al. developed a system paradigm known as MM-REACT which integrated ChatGPT with a pool of computational “experts” in order to perform vision-based tasks in a zero-shot regime [10].

Previous work has investigated the application of LLMs to EEG-related tasks. Cui et al. report NeuroGPT, an LLM-based framework used to generate and predict embeddings of EEG signal [11]. They demonstrate that this framework shows improved performance on the task of classifying motor imagery from EEG signal relative to other machine learning modalities. Other work includes the BENDR framework by Kostas et al., which applied a transformer architecture towards the task of EEG-based sleep stage classification [12].

Human EEG classification and interpretation is highly subjective, with prior studies in interrater reliability demonstrating widely variable Cohen’s kappa (κ_c) ranging from 0.3 to 0.7 amongst trained epileptologists[13]. Moreover, recent evaluations of current state-of-the-art (SOTA) EEG interpretation systems such as Persyst which utilizes deep neural networks have demonstrated subpar performance, with significant rates of false negatives and false positives raising questions about the current utility of DL EEG systems in clinical practice[14]. Here, we aim to evaluate the potential of LLMs to aid clinicians in the tasks of EEG classification and interpretation. We investigate whether an LLM-based approach may offer advantages over current DL-based EEG interpretation and classification methods in performance as well as in transparency.

2 Methods

We utilized two approaches to explore the capabilities of LLMs as they apply to clinical EEG tasks, specifically the task of classifying EEG as normal or abnormal. In both, we utilize the Temple University Hospital Abnormal Corpus, which consists of a total of 1140 hours of EEG data collected from 2993 subjects, is roughly evenly balanced between normal and abnormal recordings, and is pre-split into train and evaluation sets for consistency of evaluation across different experiments [15].

2.1 Few- and zero-shot Learning

We hypothesize that, given a relatively small amount of training data, a fine-tuned LLM (from here on referred to as EEG-GPT) will be able to classify EEG as normal or abnormal at a high level of performance relative to other machine learning modalities. We also hypothesize that with zero-shot learning, the base LLM will be able to perform significantly better than chance on the normal / abnormal classification task. Our pipeline for this experiment is shown in Figure 2.

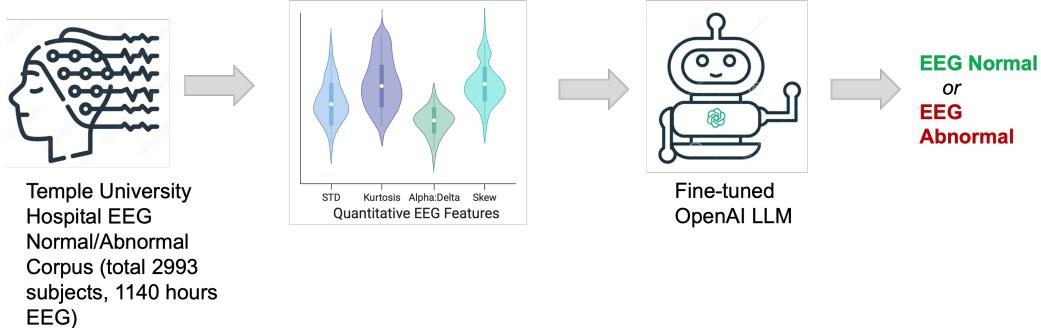


Figure 2: Pipeline for few-shot experiment

Feature selection and fine-tuning For a given EEG file, we subdivide the file into non-overlapping 20-second epochs. For each epoch, we calculate per-channel features (the subset of channels used is shown in Table 2 and the set of features used is shown in Table 1). The same set of features are calculated for each channel in order to yield a 30-feature (6 features times 5 channels) sample for each epoch. Each feature sample is labeled as normal or abnormal according to whether its parent file was labeled as normal or abnormal.

We then convert these features to a verbal representation and then use these verbal representations both to fine-tune and evaluate the da Vinci GPT-3 base LLM from OpenAI on the resulting feature set. More specifically, we use OpenAI's Completions API in order to fine-tune and evaluate EEG-GPT; to fine-tune an LLM using this API, we provide prompt and completion pair examples, with the prompt being a verbal representation of features, and the completion being the normal / abnormal label (an example prompt-completion pair is shown in Figure 2.1). OpenAI's API then fine-tunes the LLM to learn to "complete" a prompt with the corresponding completion, in this case the prediction of whether a given file is normal or abnormal.

Features calculated (over 20-second epoch)
90th percentile of voltage amplitudes
standard deviation
kurtosis
alpha:delta power ratio
theta:alpha power ratio
delta:theta power ratio

Table 1: Features calculated per channel

Channels used for feature calculation
Cz
T5
T6
O1
O2

Table 2: Channels used for feature calculation

```
{"prompt": 
  " Quantitative EEG: In a 20 second period,
  at channel Cz:[
    the 90th percentile of voltage amplitudes = 0.35 microvolts,
    standard deviation = 0.27,
    kurtosis = 0.33,
    alpha:delta power ratio = 0.22,
    theta:alpha power ratio = 9.11,
    delta:theta power ratio = 0.49];
  at channel T5:[
    the 90th percentile of voltage amplitudes = 0.10 microvolts,
```

```

    standard deviation = 0.09,
    kurtosis = 1.53,
    alpha:delta power ratio = 0.13,
    theta:alpha power ratio = 2.43,
    delta:theta power ratio = 3.13];
at channel T6:[
    the 90th percentile of voltage amplitudes = 0.18 microvolts,
    standard deviation = 0.20,
    kurtosis = 9.47,
    alpha:delta power ratio = 0.09,
    theta:alpha power ratio = 3.80,
    delta:theta power ratio = 2.95];
at channel O1:[
    the 90th percentile of voltage amplitudes = 0.34 microvolts,
    standard deviation = 0.28,
    kurtosis = 0.11,
    alpha:delta power ratio = 0.82,
    theta:alpha power ratio = 1.33,
    delta:theta power ratio = 0.92];
at channel O2:[
    the 90th percentile of voltage amplitudes = 0.30 microvolts,
    standard deviation = 0.26,
    kurtosis = 2.47,
    alpha:delta power ratio = 0.39,
    theta:alpha power ratio = 1.35,
    delta:theta power ratio = 1.89];.

Cumulative Effect Category:",

"completion":" normal"}

```

Figure 3: Example prompt-completion pair (formatted for readability)

Few- and zero-shot learning Using our derived feature set, we explore EEG-GPT’s performance on the normal/abnormal classification task in a few-shot context, and compare its performance against traditional machine learning approaches as well as against more recent deep-learning based methods. We also explore EEG-GPT’s performance in a zero-shot context – in other words, we evaluate how EEG-GPT performs on the normal/abnormal task with no fine-tuning whatsoever, both with and without in-context learning.

2.2 Evaluation of reasoning capability in EEG tool usage

We also hypothesize that an LLM-based framework will be able to effectively utilize specialized software tools to classify EEG as abnormal or normal, emulating how EEG readers systematically evaluate for signs of seizure, spikes, and/or slowing when making a classification. To develop this framework, we consider how clinical epileptologists analyze a given EEG file for abnormalities. We conceptualize this process as a decision tree, which is depicted in Figure 4. In this conceptualization, the epileptologist may first evaluate the file for evidence of seizures, then check for spikes, then check for signs of slowing – at any point, if there is strong evidence that any of these phenomena are occurring, the epileptologist can confidently claim that the file is abnormal. This is an overly simplified view of how clinical epileptologists analyze EEG, but is appropriate for initial investigations into how LLMs might utilize a similar step-wise, decision-tree-based process when coming to an analytical decision of a file.

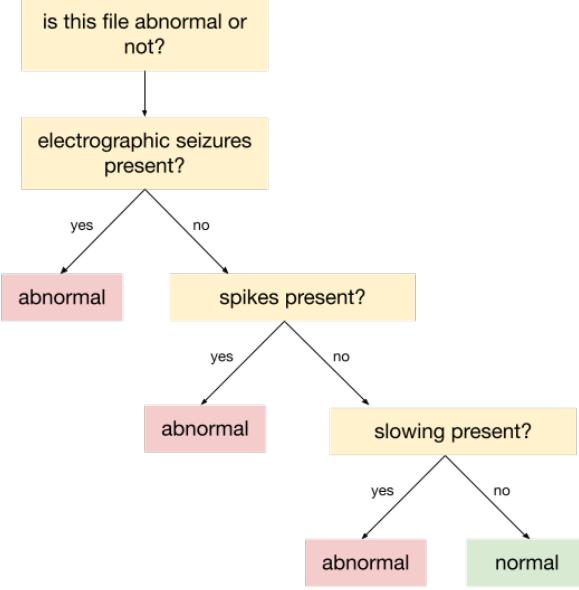


Figure 4: Simplified diagram of clinical epileptologist workflow

Tree of Thought In order to adapt LLMs to solve complex problems requiring exploration and strategic lookahead, Yao et al. developed the Tree-of-Thought framework, which allows LLMs to explore and backtrack along a decision-tree style exploration space when solving a complex problem [16]. This approach maps nicely onto our simplified decision tree depicted in Figure 4, in which a clinical epileptologist or LLM will explore the decision tree space in order to classify a file as normal or abnormal.

Specialized software tools In order to enable exploration of the decision tree we provide an agentic LLM with access to three software tools: an automated seizure detection model, an automated spike detection model, and an automated qEEG feature comparison tool. When given a particular EEG file to evaluate, the framework will explore the decision tree space, using the automated software tools available to it at each branching point, deciding whether or not to proceed based on the input from each software tool. These software tools are described in greater detail below.

Automated seizure detection model Using the Temple University Hospital seizure corpus [17], we trained an automated EEG-based seizure detection model using a convolutional neural network architecture. We ensure that there is no overlap between the subjects used to train the seizure detection model and the subjects used to evaluate the overall framework. This model takes in 10-second epochs and returns a yes/no decision as to whether a given epoch contains seizure activity or not. The model's architecture and performance on a held-out set (pulled from a set of subjects that does not overlap with the subjects used to evaluate the overall framework) are depicted below.

Automated spike detection model We use an automated spike detection algorithm based on Esteller's work investigating line length as a computational biomarker for EEG spikes [18].

qEEG feature comparison tool As a final check for abnormality, we provide the framework with a qEEG feature comparison tool. For this tool, we calculate a reference of age-based normative ranges for a subset of qEEG features (statistical moments and power ratios). Then, when given a particular EEG file, this tool calculates the set of qEEG features and calculates a cosine similarity score to the age-matched normative reference. If the cosine similarity score is higher than a particular threshold, it will return a result of "similar to a reference of normal files," and vice versa. This tool is also capable of providing a confidence score associated with its prediction based on calculating positive and negative predictive values derived from a held-out set.

3 Results

3.1 EEG-GPT demonstrates few- and zero-shot learning proficiency

We evaluate EEG-GPT in both few- and zero-shot contexts against both traditional machine learning approaches and more recent deep-learning based approaches. For the comparison against traditional machine learning approaches, we utilize the same set of training features in order to facilitate appropriate comparisons between the performance profiles of each. Comparisons against deep-learning based approaches utilize the performance metrics reported in their respective papers given that their deep learning architectures utilize the raw EEG inputs instead of the limited feature set used by EEG-GPT, and therefore their models would not be able to utilize the same feature set.

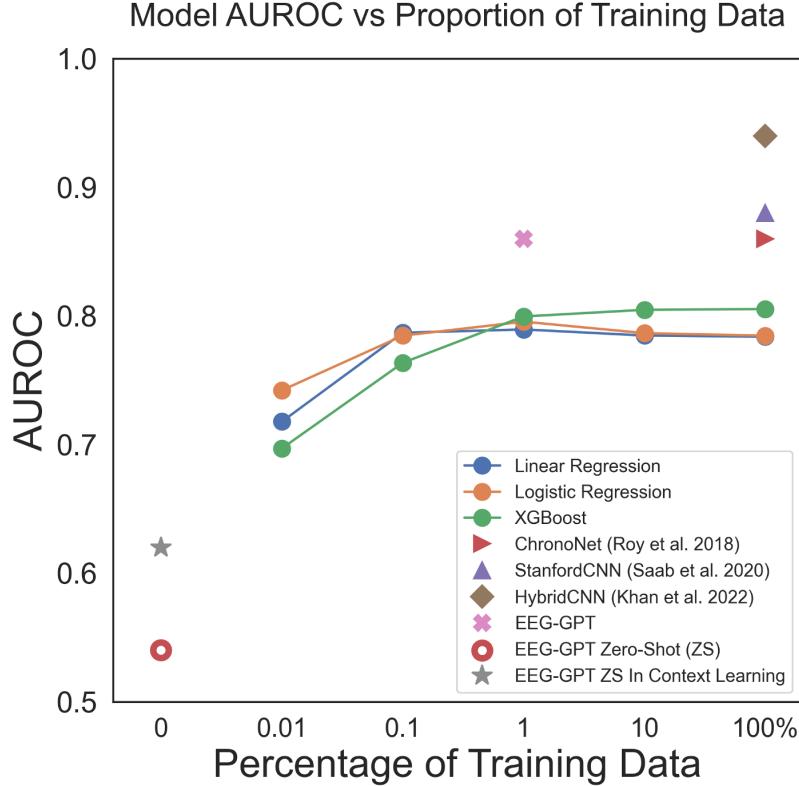
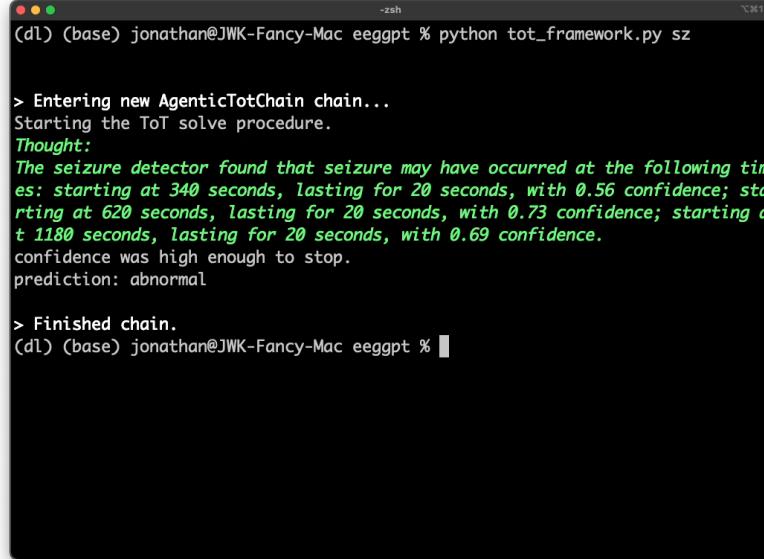


Figure 5: Resulting AUROCs of normal/abnormal classification task, plotted against proportion of training data used to fit model

When trained on 2% of available data, EEG-GPT achieves an AUROC of 0.86, as shown in Figure 5 (EEG-GPT's performance depicted with the pink "X"). This is better than traditional machine learning approaches (linear regression, logistic regression, and XGBoost), which improve performance when given increasing amounts of training data, but plateau at an AUROC of roughly 0.8. In the zero-shot context, EEG-GPT's performance remains better than chance, and improves to an AUROC of 0.63 when provided with in-context learning. When trained on 2% of available train data, EEG-GPT even matches the performance of a deep-learning based approach trained on all available data [19], although it is still outperformed by more recent approaches such as those by Saab et al. and Khan et al. [20, 21].

3.2 EEG-GPT navigates EEG tool usage with tree-of-thought reasoning

We present two exemplars of the tree-of-thought approach here. In the first demo, shown in Figure 6, we pass an EEG file which is known to contain seizures to our framework. Stepping through the solving procedure we see that the automated seizure detector immediately detects a seizure, indicating that the file is abnormal and allowing for early stoppage of the solving procedure.



```
(dl) (base) jonathan@JWK-Fancy-Mac eeggpt % python tot_framework.py sz

> Entering new AgenticTotChain chain...
Starting the ToT solve procedure.
Thought:
The seizure detector found that seizure may have occurred at the following times: starting at 340 seconds, lasting for 20 seconds, with 0.56 confidence; starting at 620 seconds, lasting for 20 seconds, with 0.73 confidence; starting at 1180 seconds, lasting for 20 seconds, with 0.69 confidence.
confidence was high enough to stop.
prediction: abnormal

> Finished chain.
(dl) (base) jonathan@JWK-Fancy-Mac eeggpt %
```

Figure 6: Framework’s analysis of EEG file known to contain seizure

In the second demo, shown in Figure 7, we pass an EEG file which is known to be abnormal yet not contain seizures to the framework. Stepping through the solve procedure, we see that the seizure detector does not find any seizures – the framework then decides to evaluate the file with another tool, the qEEG feature comparison tool. The framework then finds that the file is similar to a reference of normal EEGs, with a confidence measure of 0.54 provided by the qEEG tool itself. Presumably given the low confidence, the framework decides it needs still more information before classifying this file, and proceeds to use the automated spike detection tool – which does indeed detect spikes, indicating that this file is abnormal. At this point the framework decides it has enough information and confidence to make a final classification, and halts execution.



```
(dl) (base) jonathan@JWK-Fancy-Mac eeggpt % python tot_framework.py ab

> Entering new AgenticTotChain chain...
Starting the ToT solve procedure.
Thought:
The seizure detector did not find any times with high chance of seizure.
Thought:
This file is similar to a reference of normal EEGs, with a confidence of 0
.54.
Thought:
The following spike(s) were detected: spike lasting from time 952.61 secs to 952.72 secs, located at channels EEG FP1-REF; spike lasting from time 1164.04 secs to 1164.25 secs, located at channels EEG F4-REF, EEG C4-REF, EEG F7-REF, EEG F8-REF, EEG T3-REF, EEG A2-REF; spike lasting from time 1164.28 secs to 1164.53 secs, located at channels EEG F3-REF, EEG F4-REF, EEG C4-REF, EEG F7-REF, EEG F8-REF, EEG T3-REF, EEG A2-REF.
confidence was high enough to stop.
prediction: abnormal

> Finished chain.
(dl) (base) jonathan@JWK-Fancy-Mac eeggpt %
```

Figure 7: Framework’s analysis of abnormal EEG file known to be seizure-free

4 Discussion & Future Work

In this work we demonstrate the proficiency of EEG-GPT within a few-shot learning paradigm for EEG interpretation and classification tasks. In addition, we demonstrate a pilot exploration of EEG-GPT's reasoning ability in its capacity to navigate usage of specialist EEG tools across multiple temporal scales in a step-wise, transparent fashion. For the normal/abnormal classification task, EEG-GPT demonstrates performance that surpasses traditional machine learning methods at all amounts of training data. EEG-GPT also matches at least one previous deep-learning based approach, even when trained on 50 times less data. It is also worth noting that while all the deep learning approaches evaluated (ChronoNet, StanfordCNN, and HybridCNN) outperform or match EEG-GPT, these approaches are trained on all available training data, and utilize the raw EEG signal, which is a much richer feature space than the relatively limited set of features used by EEG-GPT. EEG-GPT's few- and zero-shot learning abilities may be partially explained by the likely presence of normative EEG data in the large data corpus used to train the base LLM.

Clinical medicine has traditionally been relatively slow to adopt machine learning techniques for clinical practice due to various ethico-legal reasons, as well as the fact that many state-of-the-art machine learning models function as "black boxes," with very little insight into how models come to a particular prediction or decision. By utilizing structured reasoning strategies such as tree-of-thought, LLMs are capable of demonstrating logical flows in their problem-solving processes. However, a significant limitation of current LLMs is their propensity to generate non-factual or "hallucinated" information[22], which limits their reliability and trustworthiness in high-stakes, clinical settings. To address this limitation, step-wise verifiability of clinical LLM systems coupled with a human in the loop or human oversight emerges as a strategy to identify LLM errors or hallucinations. This explainable AI (XAI) approach allows for monitoring and rectification of LLM system outputs, thereby mitigating the risks associated with hallucinations and easing the path to clinical adoption of AI.

Future investigation is warranted to develop and validate the potential of LLM systems for EEG interpretation and classification. Future work would involve further exploration of the clinical EEG feature space in order to further optimize performance. Given our results which seem to indicate LLMs' usefulness in performing tasks with relatively small amounts of training data, there is reason to hypothesize that LLMs may be particularly useful for rare disease applications, where training data is relatively expensive or difficult to obtain. Further study is needed in evaluating the reasoning capabilities, such as in tree-of-thought, to identify potential failure modes such as "hallucinations". In addition, as current LLMs may have poorer performance with more complex reasoning tasks[23], more work is necessary to appraise the validity of LLM logical reasoning in EEG interpretation and classification problems. Further investigation would also involve continuing to add specialized software tools to the framework and further building out the system's ability to appropriately recognize edge cases that may merit specialized software tools.

LLMs represent a promising approach for EEG interpretation and classification, and their potential to demonstrate emergent AI-reasoning abilities warrants further development and investigation. However, it is crucial that future work recognize limitations of LLMs' nascent "reasoning" capabilities especially their tendency to hallucinate. Consequently, a step-by-step verifiability with human oversight is highly recommended, as it enhances the reliability and interpretability of LLMs in clinical applications, aligning with the crucial need for trust and accountability in medical decision-making.

References

- [1] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).
- [2] Hunter Lightman et al. *Let's Verify Step by Step*. 2023. arXiv: 2305.20050 [cs.LG].
- [3] Maxwell Nye et al. *Show Your Work: Scratchpads for Intermediate Computation with Language Models*. 2021. arXiv: 2112.00114 [cs.LG].
- [4] Yaqing Wang et al. "Generalizing from a Few Examples: A Survey on Few-Shot Learning". In: *ACM Comput. Surv.* 53.3 (June 2020). ISSN: 0360-0300. DOI: 10.1145/3386252. URL: <https://doi.org/10.1145/3386252>.
- [5] Xin Liu et al. *Large Language Models are Few-Shot Health Learners*. 2023. arXiv: 2305.15525 [cs.CL].
- [6] Tianhao Li et al. *CancerGPT: Few-shot Drug Pair Synergy Prediction using Large Pre-trained Language Models*. 2023. arXiv: 2304.10946 [cs.CL].
- [7] Petter Törnberg. *ChatGPT-4 Outperforms Experts and Crowd Workers in Annotating Political Twitter Messages with Zero-Shot Learning*. 2023. arXiv: 2304.06588 [cs.CL].
- [8] Shivam Garg et al. "What Can Transformers Learn In-Context? A Case Study of Simple Function Classes". In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo et al. Vol. 35. Curran Associates, Inc., 2022, pp. 30583–30598. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/c529dba08a146ea8d6cf715ae8930cbe-Paper-Conference.pdf.

- [9] Tomaz Bratanic. *Knowledge Graphs & LLMs: Fine-Tuning vs. Retrieval-Augmented Generation*. 2023. URL: <https://neo4j.com/developer-blog/fine-tuning-retrieval-augmented-generation/>.
- [10] Zhengyuan Yang et al. *MM-REACT: Prompting ChatGPT for Multimodal Reasoning and Action*. 2023. arXiv: 2303.11381 [cs.CV].
- [11] Wenhui Cui et al. *Neuro-GPT: Developing A Foundation Model for EEG*. 2023. arXiv: 2311.03764 [cs.LG].
- [12] Demetres Kostas, Stéphane Aroca-Ouellette, and Frank Rudzicz. “BENDR: Using Transformers and a Contrastive Self-Supervised Learning Task to Learn From Massive Amounts of EEG Data”. In: *Frontiers in Human Neuroscience* 15 (2021). ISSN: 1662-5161. DOI: 10.3389/fnhum.2021.653659. URL: <https://www.frontiersin.org/articles/10.3389/fnhum.2021.653659>.
- [13] Arthur C Grant et al. “EEG interpretation reliability and interpreter confidence: a large single-center study”. In: *Epilepsy & Behavior* 32 (2014), pp. 102–107.
- [14] Taneeta Mindy Ganguly et al. “Seizure detection in continuous inpatient EEG: a comparison of human vs automated review”. In: *Neurology* 98.22 (2022), e2224–e2232.
- [15] S. Lopez. “Automated Identification of Abnormal EEGs.” In: (2017).
- [16] Shunyu Yao et al. *Tree of Thoughts: Deliberate Problem Solving with Large Language Models*. 2023. arXiv: 2305.10601 [cs.CL].
- [17] Iyad Obeid and Joseph Picone. “The Temple University Hospital EEG Data Corpus”. In: *Frontiers in Neuroscience* 10 (2016). ISSN: 1662-453X. DOI: 10.3389/fnins.2016.00196. URL: <https://www.frontiersin.org/articles/10.3389/fnins.2016.00196>.
- [18] R. Esteller et al. “Line length: an efficient feature for seizure onset detection”. In: *2001 Conference Proceedings of the 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. Vol. 2. 2001, 1707–1710 vol.2. DOI: 10.1109/IEMBS.2001.1020545.
- [19] Subhrajit Roy, Isabell Kiral-Kornek, and Stefan Harrer. “ChronoNet: A deep recurrent neural network for abnormal EEG identification”. In: *Artificial Intelligence in Medicine: 17th Conference on Artificial Intelligence in Medicine, AIME 2019, Poznan, Poland, June 26–29, 2019, Proceedings 17*. Springer. 2019, pp. 47–56.
- [20] Khaled Saab et al. “Weak supervision as an efficient approach for automated seizure detection in electroencephalography”. In: *NPJ digital medicine* 3.1 (2020), p. 59.
- [21] Imran Ullah Khan, Sitara Afzal, and Jong Weon Lee. “Human activity recognition via hybrid deep learning based model”. In: *Sensors* 22.1 (2022), p. 323.
- [22] Lei Huang et al. “A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions”. In: *arXiv preprint arXiv:2311.05232* (2023).
- [23] Ruocheng Wang et al. “Hypothesis search: Inductive reasoning with language models”. In: *arXiv preprint arXiv:2309.05660* (2023).

Enhancing EEG-to-Text Decoding through Transferable Representations from Pre-trained Contrastive EEG-Text Masked Autoencoder

Jiaqi Wang^{1,2}, Zhenxi Song^{1*}, Zhengyu Ma², Xipeng Qiu³, Min Zhang¹, Zhiguo Zhang^{1,2†}

¹School of Computer Science and Technology, Harbin Institute of Technology Shenzhen, China

² Peng Cheng Laboratory, China

³School of Computer Science, Fudan University, China

{songzhenxi, zhiguo Zhang}@hit.edu.cn

Abstract

Reconstructing natural language from non-invasive electroencephalography (EEG) holds great promise as a language decoding technology for brain-computer interfaces (BCIs). However, EEG-based language decoding is still in its nascent stages, facing several technical issues such as: 1) Absence of a hybrid strategy that can effectively integrate cross-modality (between EEG and text) self-learning with intramodality self-reconstruction of EEG features or textual sequences; 2) Under-utilization of large language models (LLMs) to enhance EEG-based language decoding. To address above issues, we propose the **Contrastive EEG-Text Masked Autoencoder (CET-MAE)**, a novel model that orchestrates compound self-supervised learning across and within EEG and text through a dedicated multi-stream encoder. Furthermore, we develop a framework called **E2T-PTR** (**EEG-to-Text** decoding using **P**retrained **T**ransferable **R**epresentations), which leverages pre-trained modules alongside the EEG stream from CET-MAE and further enables an LLM (specifically BART) to decode text from EEG sequences. Comprehensive experiments conducted on the popular text-evoked EEG database, ZuCo, demonstrate the superiority of E2T-PTR, which outperforms the baseline framework in ROUGE-1 F1 and BLEU-4 scores by 8.34% and 32.21%, respectively. Our proposed pre-trained EEG-Text model shows the potential to improve downstream tasks involving EEG and text. This opens up promising avenues for its application in inner speech BCI paradigms, meriting further investigation.

1 Introduction

Decoding natural language from non-invasive brain recordings with electroencephalography (EEG) is an emerging topic that holds promising benefits

*Corresponding author

†Corresponding author

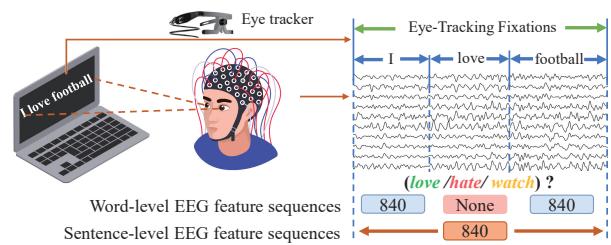


Figure 1: **Text-evoked EEG Recording in ZuCo datasets.** Participants' EEG and eye-tracking data are simultaneously recorded during natural reading to capture text-evoked brain activity.

for patients suffering from cognitive impairments or language disorders. Thanks to the burgeoning development of pre-trained large language models (LLMs) (Zhao et al., 2023a), the potential of using an open vocabulary to decode human brain activity has been gradually unlocked. Specifically, through the commendable text understanding and generation capabilities of cutting-edge LLMs (Touvron et al., 2023; Ouyang et al., 2022), translating complex spatio-temporal EEG signals into nuanced textual representations, which is known as EEG-to-Text, is being achieved. Compared to conventional paradigms of brain-computer interfaces (BCIs), such as motor imagery (MI) (Al-Saegh et al., 2021), steady-state visual evoked potential (SSVEP) (Wang et al., 2016), and P300 (Cecotti and Graser, 2010), EEG-to-Text can convey much more intended commands from the human brain to computers, and thus presents a more extensive range of applications. Its potential as a novel and powerful BCI paradigm suggests it could contribute to advancements in the field of imagined or inner speech BCIs.

Several existing EEG studies (Li et al., 2022a; Yi et al., 2024) were focused on developing specialized pre-trained models for EEG only, aiming to extract universal semantic representations from the human brain. However, the pre-trained model

bridging EEG and text has been ignored, which may be important to enhance the representation learning for inter-modality conversion (Bai et al., 2023). This motivates us to develop a hybrid model to orchestrate compound pre-trained representations across and within EEG and text. This endeavor faces the core challenge: *How to bridge the semantic gap between EEG and text while establishing an implicit mapping in the latent representation space?* Responding to this challenge, we focus on self-supervised learning (SSL), because of its great capability in multi-modal representation learning (Chen et al., 2024). Contrastive learning is one of the important SSL strategies, learning semantic-level representations across modalities (as CLIP does for language and image) (Radford et al., 2021). Masked modeling methods exhibit significant capability of intra-modality self-reconstruction, such as BERT (Devlin et al., 2019) in nature language processing and masked autoencoder (MAE) (He et al., 2022) in computer vision.

Inspired by the above prevailing SSL strategies, we propose a novel pre-trained model to align EEG and text, Contrastive EEG-Text Masked Autoencoder (CET-MAE), as shown in Figure 2(a). CET-MAE integrates contrastive learning and masked signal modeling through a dedicated multi-stream encoder. It effectively learns pre-trained representations of EEG and text by balancing the latent embeddings represented by self-reconstruction and the semantic-level aligned embeddings of text tokens and text-evoked EEG features. In terms of masked signal modeling, CET-MAE implements a high mask ratio (specifically, 75%) on both EEG and text data, presenting a meaningful challenge for the model to handle an increased amount of missing information during the reconstruction phase. This setting not only enhances the model’s understanding of individual modality but also facilitates cross-modal interactions and support.

Furthermore, to make the most of LLMs’ capability in language understanding and generation as well as to fully use pre-trained representations learned by CET-MAE, we introduce a new EEG-to-Text decoding framework, EEG-to-Text using Pre-trained Transferable Representations (E2T-PTR). E2T-PTR utilizes pre-trained modules alongside the EEG stream from CET-MAE and further adopts the BART (Lewis et al., 2020) to decode language from EEG sequences. By transferring the pre-trained representations from CET-MAE, E2T-PTR significantly enhances EEG-to-Text decoding,

surpassing both the baseline and state-of-the-art (SOTA) methods.

Our main contributions are summarised below:

- Introducing CET-MAE, the first pre-trained EEG-text model for EEG-based language decoding. CET-MAE integrates the reconstruction of text and EEG features with semantic alignment, forming a multi-stream SSL framework for both intra-modality and cross-modality representation learning.
- Developing a new EEG-to-Text framework via E2T-PTR. The new E2T-PTR framework can leverage CET-MAE’s pre-trained EEG representations and the capabilities of LLMs (BART) for text generation.
- Conducting extensive EEG-to-Text experiments on three, four, and five reading tasks in ZuCo. Results demonstrate that our framework outperforms previous works, offering valuable insights into leveraging pre-trained transferable representations to enhance EEG-to-text decoding.

2 Related Works

2.1 Self-supervised Representations Learning

Multimodal self-supervised representation learning aims to explore the interactions between different modalities to produce semantically generalizable representations for downstream tasks.

In recent years, there have been substantial progresses across various modalities, such as vision-language pre-training (Zhao et al., 2023b; Lin et al., 2023). A range of existing methods rely on contrastive learning, which can effectively draw closer to the global representations of matched pairs in latent spaces with semantic-level self-supervised constraints. But contrastive learning sometimes tends to overlook the self-information of individual modalities, particularly at more granular levels. On the other hand, multimodal masked signal modeling integrates cross-modality self-learning with intra-modality self-reconstruction, focusing on reconstructing one modality from another. This approach may help the model learn the associations between modalities. However, it may lead to an excessive emphasis on fine-grained details, potentially weakening the overall cross-modality correlation and causing issues such as insensitivity to whether the inputs are matched pairs. A series of

recent works, such as CMAE (Huang et al., 2023), CAV-MAE (Gong et al., 2022) and SimVTP (Ma et al., 2022), have already successfully integrated both contrastive learning and masked signal modeling so that their complement advantages can be utilized.

Our work draws inspiration from the above SSL methods but with a novel strategy. In the proposed CET-MAE, the utilization of both text and EEG streams not only achieves an explicit contrastive learning objective to capture global coordination but also avoids erroneous learning processes. Meanwhile, the utilization of the joint stream can facilitate the information interaction between modal-specific embeddings to achieve masked signal modeling effectively. To the best of our knowledge, this is the first EEG-to-Text masked autoencoder that attempts to establish transferable representation learning between EEG and text.

2.2 Open Vocabulary EEG-to-Text Decoding

Previous works (Nieto et al., 2022; Kamble et al., 2023) on EEG-to-Text have been severely confined by a limited number of (several or tens of) words in terms of vocabulary size. These closed-vocabulary efforts primarily focused on recognizing low-level linguistic features, such as individual words or syllables. However, these works can hardly capture more complex, high-level semantic and contextual aspects of language.

The development of LLMs has significantly enhanced the field of EEG-based text decoding. The first work using LLM (Wang and Ji, 2022) integrates an additional EEG encoder to align the pre-trained BART for EEG-to-Text, providing important inspiration for subsequent works. C-SCL (Feng et al., 2023) employs curriculum learning to effectively mitigate the discrepancy between subject-dependent and semantic-dependent EEG representations in EEG-to-Text translation. De-Wave (Duan et al., 2024) uses a quantized variational encoder to convert continuous EEG signals into discrete sequences, alleviating the reliance on eye fixations. Despite advancements, prior efforts struggled to bridge the complex semantic gap between EEG and text on an open-vocabulary scale. Our proposed CET-MAE aims to tackle this challenge. Additionally, our E2T-PTR framework transfers CET-MAE’s representations and leverages the BART to achieve superior text generation outcomes.

3 Methods

3.1 Preliminary

ZuCo benchmark dataset. For our work, we use the ZuCo1.0 (Hollenstein et al., 2018) and ZuCo2.0 (Hollenstein et al., 2023) datasets, which contain the EEG and eye tracking data during five natural reading tasks. The corpus for sentiment reading (SR) task v1.0 comes from the movie reviews. The corpus for the remaining four tasks is sourced from Wikipedia and comprises two versions each of Natural Reading (NR) and Task-Specific Reading (TSR), specifically NR v1.0, NR v2.0, TSR v1.0, and TSR v2.0. The word-level EEG was recorded and aligned by the eye-tracking fixations, and the sentence-level EEG was recorded during the entire reading procedure. We follow the preprocessing and dataset splits established by baseline work (Wang and Ji, 2022).

Natural masking ratios of EEG feature sequences. Our investigation reveals the word-level contextual EEG presentations in ZuCo datasets are severely corrupted due to missing eye-tracking fixations, leading to mismatches between EEG raw data and text, as shown in Figure 1. This misalignment leads to fragmented word-level EEG feature sequences, which fails to capture the cohesive semantics of entire sentences and inevitably complicates the representations learning of EEG and text.

Different from previous works, we concatenate the word-level EEG features and the sentence-level EEG features as our EEG feature sequences E as

$$E = [E_{word1}, E_{word2}, \dots, E_{wordN}, E_{sentence}]. \quad (1)$$

Incorporating sentence-level EEG features offers several benefits. First, it provides a holistic view of EEG sequences, enriching the interpretation of overall sentence semantics. Secondly, it acts as a form of data augmentation, which can mitigate the issue of data incompleteness, thereby alleviating semantic discrepancies caused by the misalignment between word-level EEG and text. To provide a clearer overview, we have presented the detailed statistics of the natural masking ratio (NMR) of EEG feature sequences under three categories of reading task combinations in Appendix A.

Definitions in EEG-to-Text Decoding. Given a sequence of EEG features E as the input to the model M , the aim is to decode the ground-truth word tokens W from open-vocabulary V via M . These corresponding EEG-Text pairs $\langle E, W \rangle$ are collected during natural readings.

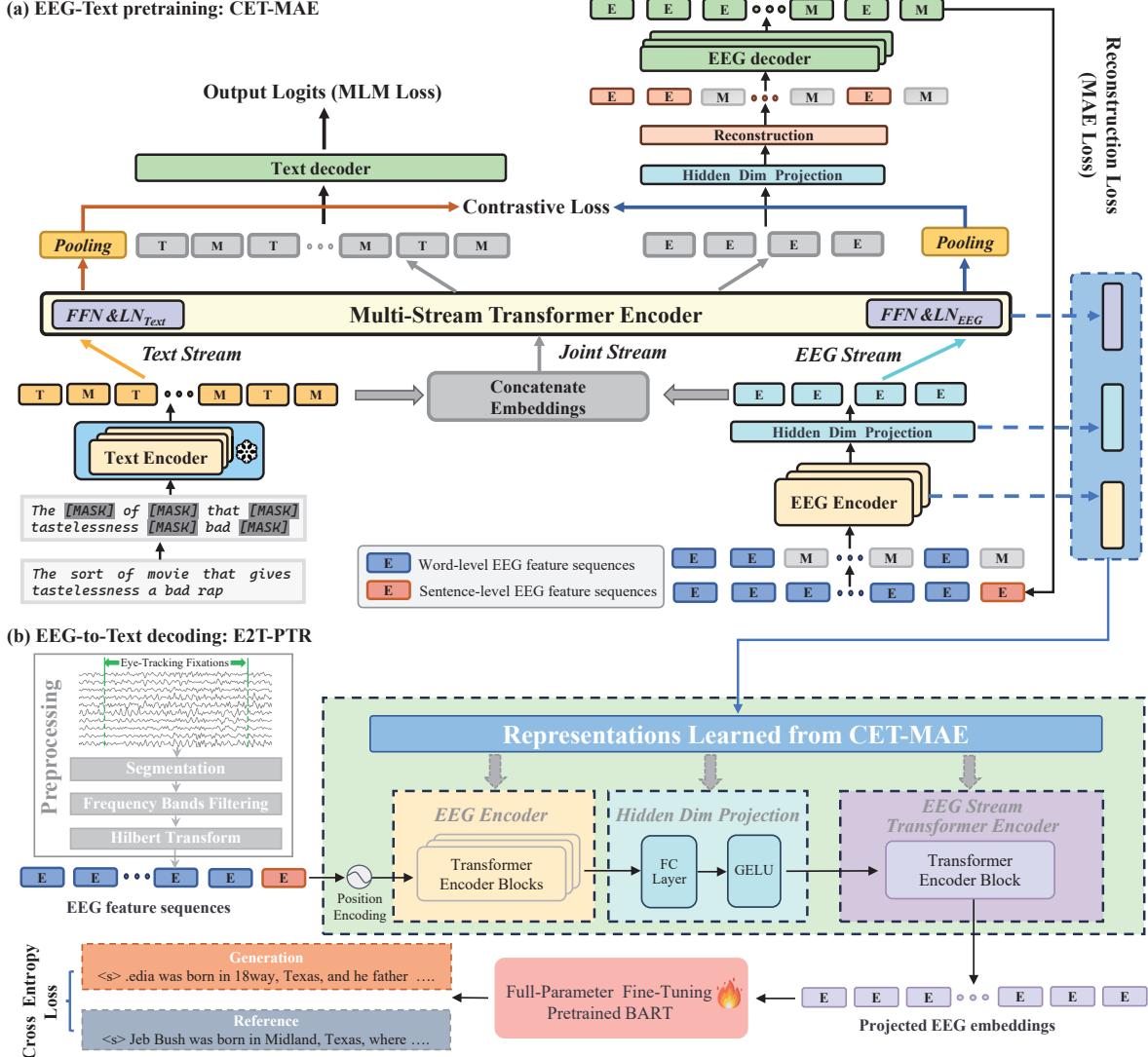


Figure 2: Illustration of the proposed EEG-text pre-training model (CET-MAE) and EEG-to-Text decoding framework (E2T-PTR). (a) **CET-MAE Model:** CET-MAE features modality-specific autoencoders with a masking strategy for text and EEG features, complemented by a multi-stream transformer encoder that orchestrates self-reconstruction and cross-modality semantic alignment, enhancing representation learning for EEG semantic decoding. (b) **E2T-PTR Framework:** E2T-PTR transfers both word- and sentence-level EEG representations extracted from CET-MAE’s pre-trained modules, further facilitating text generation through the BART.

During the testing phase, the model M operates with an implicit understanding of the ground-truth word tokens W . Its primary objective remains to decode the EEG feature sequences E to generate an output that closely matches tokens W . This involves the model generating the sequence of words with the highest probability within the probability distribution P of the V .

3.2 EEG-Text Masking

We perform random masking on the text tokens, followed by processing with BERT. For EEG masking, we adopted the following settings. Word-level EEG feature sequences are randomly masked, while

sentence-level EEG feature sequences are compulsorily masked. This aims to force the model to fully reconstruct the contextual semantics within the sentence-level EEG feature sequences.

3.3 CET-MAE Encoder

As illustrated in Figure 2(a), the CET-MAE model needs to extract the embeddings of text and EEG separately and then feed the embeddings into the multi-stream transformer encoder to learn the cross-modal representations.

Text encoder. We utilize the pre-trained encoder-decoder model BART as the text encoder. Due to the suitable capabilities in natural language

understanding and generation (Li et al., 2022b), we opt to freeze weights of the BART¹ encoder to maintain its high-level language comprehension from the last hidden states. Firstly, the text tokens are converted into high-quality text embeddings with positional encoding by BART. The learnable embeddings are then used to replace the masked word tokens.

EEG encoder. The EEG encoder is designed as a Multi-layer Transformer Encoder (Vaswani et al., 2017) to capture the temporal relationships from EEG sequences with spatial and frequency features in each token. A learnable linear projection layer is employed to transform the EEG embeddings from the EEG encoder, aligning their dimensions with those of the text embeddings.

Multi-stream Transformer encoder. The pivotal design of this module lies in the integration of EEG, text, and the joint streams. We implement the dual-modality streams for EEG-text contrastive learning, especially using a specialized head for each modality. It is equipped with the layer normalization (LN) and the feed-forward network (FFN) enabling the production of embeddings that preserve their unique properties (Gong et al., 2022). Notably, we control the learning process to ensure that learnable vectors at masked positions do not enter into the text stream, thereby preventing the inclusion of misleading contrastive feedback. Equally crucial for the two reconstruction tasks, the joint stream is utilized to facilitate the integration of the embeddings from both text and EEG modalities. This design aims to deepen the interaction and enhance the cooperation between EEG and text, fostering a more effective learning synergy.

3.4 CET-MAE Decoder

We apply a lightweight Transformer encoder as the EEG decoder. For EEG reconstruction tasks, EEG embeddings are first mapped to the original dimensions through a learnable linear projection layer. Subsequently, EEG embeddings with learnable masked tokens are inserted back into their original positions. The final EEG embeddings added to the positional embeddings are fed into the EEG decoder. Since the text encoder has already encoded the masked tokens and captured their positional information within the text, we employ a learnable linear projection layer as the text decoder to predict the masked text tokens.

¹<https://huggingface.co/facebook/bart-large>

3.5 CET-MAE Training Objectives

CET-MAE is pre-trained by three objectives: (1) Masked Text Modeling (L_T): it aims to predict the masked text tokens by utilizing hybrid representations that integrate information from both textual and EEG embeddings. (2) Masked EEG Modeling (L_E): it learns to reconstruct the original EEG feature sequences, especially predicting masked word- and sentence-level features based on hybrid representations, where the error is measured by mean square error (MSE). (3) EEG-Text Contrastive Learning (L_{CL}): it involves a process where the corresponding EEG and text representations are computed by separate global average pooling layers. The objective is to bring the aligned pairs (matched EEG and text embeddings) closer together while pushing unpaired ones further apart. Our goal L is minimizing is the summation of these three learning objectives:

$$L = \lambda_T \cdot L_T + \lambda_E \cdot L_E + \lambda_{CL} \cdot L_{CL} \quad (2)$$

3.6 E2T-PTR Framework

The proposed E2T-PTR is illustrated in Figure 2(b). It can be summarized into the following key points.

Word-sentence level input tokens. We add the sentence-level EEG features as our input tokens. As detailed in 3.1, concatenating the sentence-level EEG feature sequences as the last token can effectively alleviate the incoherent contextual semantics due to gaps in word-level EEG features.

Effective transfer capability. We investigate how to effectively transfer the cross-modality representations learned from the CET-MAE to downstream tasks such as EEG-to-Text decoding. The E2T-PTR employs a synergy of the following critical components: the EEG encoder, the linear projection layer, and the EEG-stream transformer encoder, all of which are integral components as outlined within the CET-MAE. For the LLM backbone, we also apply the BART which excels at natural language generation tasks.

Fine-tuning strategy. We fine-tune all parameters of E2T-PTR during the training phase. The weights of CET-MAE are first loaded into the EEG encoder, the linear projection layer, and the EEG-stream transformer encoder. As the linguistic backbone of E2T-PTR, the BART is also fully fine-tuned to improve its ability to generate fine-grained text tokens from EEG embeddings.

Method	Training Sample	BLEU-N(%)				ROUGE-1(%)		
		N=1	N=2	N=3	N=4	P	R	F
EEG2Text (Wang and Ji, 2022)	10710	40.1	23.1	12.5	6.8	31.7	28.8	30.1
DeWave (Duan et al., 2024)	10710	41.35	24.15	13.92	8.22	33.71	28.82	30.69
E2T-PTR (proposed)	10710	42.09	25.13	14.84	8.99	35.86	30.01	32.61
C-SCL (Feng et al., 2023)	14567	35.91(—)	25.91(—)	21.31(—)	18.89(—)	—	—	—
C-SCL*	14407	34.87(44.14)	25.32(31.61)	21.17(25.67)	18.98(22.51)	36.97	34.31	35.51
E2T-PTR (proposed)	14407	34.92(44.31)	25.43(31.67)	21.00(25.52)	18.59(22.22)	37.15	33.93	35.39
EEG2Text*	18791	58.06	49.98	46.21	44.13	52.31	48.76	50.41
E2T-PTR (proposed)	18791	59.20	50.77	46.82	44.63	53.76	50.03	51.77

Table 1: Comparison of our E2T-PTR framework with previous methods on the ZuCo dataset for three and four reading tasks. * means that our reproduced results. Results enclosed in parentheses are calculated following the approach of EEG2Text, which includes retaining consecutive repeated words in the generated text.

	Ground Truth: He was first <i>appointed</i> to fill the <u>Senate seat</u> of Ernest Lundeen who had died in office.
(1)	EEG2Text: was a <i>elected</i> to the the <u>position seat</u> in the Hemy in died died in 18 in
	E2T-PTR: was the <i>elected</i> to the the <u>position seat of John Hemy</u> , resigned <u>resigned</u> in office .
	Ground Truth: Jeb Bush was born in <u>Midland, Texas</u> , where his father was running an oil drill company .
(2)	DeWave: <u>uan Bush</u> was a in 18way, Texas , in he father was an insurance refinery company .
	E2T-PTR: <u>uan Bush</u> was born in <u>Newway, Texas</u> , and his father was a a insurance company company .
	Ground Truth: After Raymond graduated from high school , he enrolled in the "Universidad del Sagrado Corazon" (University of the Sacred Heart) of San Juan, where he earned a <u>Bachelors Degree</u> ...
(3)	E2T-PTR: the's from Yale school , he went in the <u>UniversityAmericancities de Reynado Corazon"</u> (University of the Sacred Heart) in Spain Francisco, Puerto he studied a <u>Bachelor.ors</u> ...

Table 2: EEG-to-Text decoding results. **Bold** words indicate exact match, *Italic* words indicate semantic resemblance, and Underline words indicate error match. We evaluate the translation performance of the same test sentences reported in EEG2Text, DeWave.

4 Experiments

4.1 Datasets and Evaluation

We pre-trained our CET-MAE models under three, four, and five reading tasks in ZuCo v1.0 and ZuCo v2.0. For fairness, we assessed the performance of E2T-PTR for the EEG-to-Text task under the identical dataset scale used during the pre-training phase. We adopt the BLEU and ROUGE-1 scores for evaluating the EEG-to-Text generation performance. More details are presented in Appendix B.

4.2 Implementation Details

The CET-MAE model features a robust EEG encoder with transformer encoder blocks (6 layers, 2048 hidden dimensions, and 8 attention heads). The EEG decoder is a lightweight transformer encoder of 1 layer with 8 heads. The multi-stream transformer encoder is designed with 1 layer, a 4096 hidden dimension, and 16 attention heads. The mask ratios for EEG feature sequences and textual tokens are set at 75% (which can achieve the best results based on trial-and-error). For the CET-

EEG Mask Ratio (%)	Text Mask Ratio (%)	BLEU-N (%)			
		N=1	N=2	N=3	N=4
25	25	42.14	25.02	14.55	8.62
50	25	41.74	24.75	14.39	8.52
50	50	41.80	24.69	14.25	8.40
75	50	41.93	25.02	14.72	8.81
75	75	42.09	25.13	14.84	8.99

Table 3: The performance of our E2T-PTR framework under different combinations of CET-MAE mask ratios rising from 25% to 50%, and to 75% across three reading tasks.

MAE pertaining objective L , we set $\lambda_T=0.1$, $\lambda_E=1$, $\lambda_{CL}=0.01$. This setting is refined through experiments to balance the gradients of each loss in the overall training objective, ensuring that the model learns effectively from each task. We pre-train the CET-MAE model from scratch for 100 epochs. Subsequently, we fine-tune the E2T-PTR model for EEG-to-Text tasks over 50 epochs, employing a batch size of 32 and utilizing the AdamW optimizer. More details are provided in Appendix B.

4.3 Main Results

Table 1 shows the performance of our E2T-PTR framework on the ZuCo benchmarks. In three reading tasks, E2T-PTR achieves BLEU-1 to BLEU-4 SOTA scores of 42.09%, 25.13%, 14.84%, and 8.99%, respectively. Moreover, it outperforms best in ROUGE-1 Precision, Recall, and F1 scores compared to recent works. Notably, without removing repetitive generated word tokens, E2T-PTR surpasses C-SCL in BLEU-1 and BLEU-2 scores across four reading tasks. Particularly under the five reading tasks with 18791 training samples, E2T-PTR scores 59.20%, 50.77%, 46.82%, and 44.63% in BLEU-1 to BLEU-4, significantly exceeding the baseline work EEG2Text.

Table 2 presents a comparative analysis of the decoding results between our model and other models under three reading tasks. Our model E2T-PTR demonstrates an enhanced ability to generate more complete grammatical structures, which is evident from the reduced error rates and increased semantic coherence in the decoded sentences, exemplified by expressions such as “**his father was**” and “**Bush was born in**”. Our model also excels in decoding common and proper nouns, such as “**office**” and “**University of the Sacred Heart**”. It also adeptly produces semantically similar words, such as, “*appointed*” vs “*elected*”, and “*Ernest Lundeen*” vs “*John Hemy*”. Intriguingly, upon expanding our training samples to 1.75 times (10710 to 18791), we observe an obvious improvement in the translation quality of the model, especially concerning fine-grained recognition. More comprehensive results are included in the Appendix C.

Our investigation delved into the transfer performance of CET-MAE across varying EEG and text masking ratios under three reading tasks. Table 3 details the performance shifts under different combinations of masking ratios rising from 25% to 50%, and to 75%. We discovered that the CET-MAE model excels at the higher masking ratios of 75%, starkly contrasting with the traditional 15% mask ratio suggested in BERT. This result is consistent with recent findings in multi-modal masked models (Ma et al., 2022; Geng et al., 2022), suggesting that inter-modal interactions may promote performance improvement. We further ponder this phenomenon and suggest that, in terms of CET-MAE structure, it appears to be suited for reconstructing masked EEG features and predicting masked word tokens. In terms of the masking strat-

egy, forcefully masking sentence-level EEG embeddings can better compel the model to learn global semantic information. Furthermore, we discuss the overall masking ratio for the EEG, the natural EEG masking ratio under three reading tasks is 32.51% as mentioned in Appendix A. Therefore, the total masking ratio for the EEG is 83.13%² (32.51% of natural + 50.62% of CET-MAE masked).

For a more rigorous validation, we further implemented the leave-one-subject-out validation strategy for both the CET-MAE model and the E2T-PTR framework, detailed in Table 4. This validation approach proved extremely valuable in testing the generalization performance across different subjects within the EEG dataset. Given the inherent noise and individual variability in EEG data, it is crucial to evaluate how well a model performs under such conditions. The results obtained from the leave-one-subject-out validation not only exceeded our initial performance metrics presented in Table 1 but also underscored the strong generalizability of our models. These results affirm the ability of our models to effectively manage the inherent variability in EEG data, thereby demonstrating robust performance as each subject’s data was sequentially excluded from the training set.

4.4 Ablation Studies

Table 5 details the ablation experiments, affirming the effectiveness of each component in our approaches for EEG-to-Text generation quality. First, sentence-level EEG features positively impact BLEU scores, notably BLEU-1, underscoring their importance in capturing essential semantic information for improved text generation. Second, CET-MAE, focusing on masked signal modeling and contrastive learning between EEG and text, is fundamental. Integrating CET-MAE with the baseline framework (Wang and Ji, 2022) significantly boosts BLEU scores, especially BLEU-4. Third, combining E2T-PTR with CET-MAE enhances performance across metrics, particularly Precision, Recall, and F1 score of ROUGE-1, showcasing E2T-PTR’s role in effectively transferring CET-MAE’s learned representations.

4.5 Transfer Performance of SSL Models

We further pre-train and compare the transfer performance of the following SSL models: 1) Contrastive EEG-Text (CET) learning model: The CET

²Overall Masking Ratio = NMR + (1 - NMR) × CET-MAE Masking Ratio.

Model	Validation Strategy	BLEU-N(%)				ROUGE-1 (%)		
		N=1	N=2	N=3	N=4	P	R	F
E2T-PTR	Split each subject’s data in an 8:1:1 ratio	42.09	25.13	14.84	8.99	35.86	30.01	32.61
	Leave-one-subject-out Cross-validation	44.98	27.57	17.23	10.99	38.74	31.84	34.82

Table 4: Performance comparison of E2T-PTR frameworks between two different data splitting strategies under three reading tasks and used BLEU-N (%) and ROUGE-1 (%) as the evaluation metrics.

Sentence-level EEG feature sequences	CET-MAE	E2T-PTR	Training Sample	BLEU-N (%)				ROUGE-1 (%)		
				N=1	N=2	N=3	N=4	P	R	F
✗	✗	✗	10710	41.16	23.99	13.49	7.68	34.68	28.96	31.45
✓	✗	✗	10710	41.63	24.48	13.96	8.06	35.13	29.27	31.83
✓	✓	✗	10710	41.88	24.85	14.52	8.74	35.26	29.50	32.02
✓	✓	✓	10710	42.09	25.13	14.84	8.99	35.86	30.01	32.61

Table 5: The results of ablation experiments on CET-MAE and E2T-PTR structures under three reading tasks. We verified the effectiveness of each component and used BLEU-N (%) and ROUGE-1 (%) as the evaluation metrics.

Metrics (%)	Our SSL Models		
	CET	ET-MAE	CET-MAE
BLEU-1	41.77	41.80	42.09
BLEU-2	24.68	24.72	25.13
BLEU-3	14.33	14.43	14.84
BLEU-4	8.60	8.53	8.99
ROUGE-1 P	35.59	35.06	35.86
ROUGE-1 R	30.11	29.31	30.01
ROUGE-1 F	32.51	31.82	32.61

Table 6: Evaluating transfer performance across CET, ET-MAE, and CET-MAE under three reading tasks.

that has no reconstruction objective. For a fair comparison, we implement CET using the same encoder architecture (modal-specific encoders + multi-stream encoder) with CET-MAE but remove the reconstruction task (L_E and L_T). We use this model to investigate the impact of contrastive learning. 2) EEG-text masked autoencoder (ET-MAE) model: The ET-MAE has the same architecture as CET-MAE but the contrastive loss (L_{CL}) is set to 0. The masking strategy is the same as CET-MAE. We use this model to examine the effectiveness of masked signal modeling. 3) Our proposed CET-MAE is detailed in Section 3.

To ensure fairness, CET and ET-MAE are pre-trained with the same pipeline as CET-MAE. We assess their EEG-to-Text transfer performance using the E2T-PTR framework. Results in Table 6 demonstrate CET-MAE’s superiority over two other SSL models (CET and ET-MAE) across most evaluation metrics. Specifically, CET-MAE achieves

improvements of 0.32%, 0.45%, 0.51%, and 0.39% in BLEU-1 to BLEU-4, respectively, compared to CET. Against ET-MAE, CET-MAE records increases of 0.29%, 0.41%, 0.41%, and 0.46% for these metrics, respectively. The trend of enhancement is consistent in ROUGE-1 metrics as well.

5 Conclusion

This study contributes to the development of EEG-based language decoding by introducing an effective EEG-text pre-trained model, CET-MAE, and a highly capable and LLM-empowered EEG-to-Text decoding framework, E2T-PTR. CET-MAE uses a multi-stream architecture to incorporate both intra- and cross-modality SSL within one unified system: 1) Intra-modality streams explore representative embeddings that reflect the intrinsic characteristics of EEG or text sequences, leveraging masked modeling with a mask ratio of up to 75%; 2) Inter-modality stream provides dual-modal representations to enhance intra-modality reconstruction and constrains the encoder to maximize semantic consistency between text and its corresponding EEG sequences. E2T-PTR transfers pre-trained EEG representations and leverages BART’s capabilities for text generation from these consistent and representative features. Extensive experiments on the latest text-evoked EEG dataset, ZuCo, demonstrate the superiority of this work in both qualitative and quantitative assessments. The proposed CET-MAE model shows great potential for enhancing EEG-based language decoding tasks and could be utilized for other inner speech BCI datasets.

Limitation

The limitations of our study are summarized as follows:

Dataset Scale: The performance of both the CET-MAE model and the E2T-PTR framework is constrained by the scale of currently available datasets. We are in the process of developing our datasets to fully exploit the potential of our models and frameworks.

Teacher Forcing: While our results are pushing the open vocabulary EEG-to-Text decoding performances to a new SOTA, they still depend on the implicit use of teacher forcing, a common precondition in recent studies (Wang and Ji, 2022; Duan et al., 2024; Feng et al., 2023; Xi et al., 2023). This reliance on teacher forcing could be constraining the full capabilities of the LLMs. Noted that recent work (Yang et al., 2024) has reported promising results with the autoregressive capabilities of large speech models like Whisper (Radford et al., 2023) on the MEG datasets (Schoffelen et al., 2019). This may offer potential solutions to the challenges of using teacher forcing in the EEG-to-Text field. Our future work will aim to verify the correctness of the aforementioned new methods and explore the autoregressive capabilities of LLMs to reduce reliance on teacher forcing.

Exploration of LLMs: We plan to explore more advanced LLMs to enhance our EEG-to-Text decoding capabilities. This will involve testing new models and techniques to improve performances and uncover deeper insights from EEG data.

Ethics Statement

In this work, we do not generate new EEG data, nor do we perform experiments on human subjects. We use the publicly available ZuCo v1.0 and ZuCo v2.0 datasets without any restrictions. We do not anticipate any harmful applications of our work.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 62306089) and the China Post-doctoral Science Foundation (Nos. 2023M730873 and GZB20230960).

References

- Ali Al-Saegh, Shefa A. Dawwd, and Jassim M. Abdul-Jabbar. 2021. Deep learning for motor imagery EEG-based classification: A review. *Biomedical Signal Processing and Control*, 63:102172.

Yunpeng Bai, Xintao Wang, Yan-pei Cao, Yixiao Ge, Chun Yuan, and Ying Shan. 2023. Dreamdiffusion: Generating high-quality images from brain eeg signals. *arXiv preprint arXiv:2306.16934*.

Hubert Cecotti and Axel Graser. 2010. Convolutional neural networks for p300 detection with application to brain-computer interfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):433–445.

Xiaokang Chen, Mingyu Ding, Xiaodi Wang, Ying Xin, Shentong Mo, Yunhao Wang, Shumin Han, Ping Luo, Gang Zeng, and Jingdong Wang. 2024. Context autoencoder for self-supervised representation learning. *International Journal of Computer Vision*, 132(1):208–223.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186.

Yiqun Duan, Charles Chau, Zhen Wang, Yu-Kai Wang, and Chin-teng Lin. 2024. Dewave: Discrete encoding of eeg waves for eeg to text translation. *Advances in Neural Information Processing Systems*, 36.

Xiachong Feng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2023. Aligning semantic in brain and language: A curriculum contrastive method for electroencephalography-to-text generation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*.

Xinyang Geng, Hao Liu, Lisa Lee, Dale Schuurmans, Sergey Levine, and Pieter Abbeel. 2022. Multimodal masked autoencoders learn transferable representations. *arXiv preprint arXiv:2205.14204*.

Yuan Gong, Andrew Rouditchenko, Alexander H Liu, David Harwath, Leonid Karlinsky, Hilde Kuehne, and James Glass. 2022. Contrastive audio-visual masked autoencoder. *arXiv preprint arXiv:2210.07839*.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009.

Nora Hollenstein, Jonathan Rotsztejn, Marius Troendle, Andreas Pedroni, Ce Zhang, and Nicolas Langer. 2018. Zuco, a simultaneous eeg and eye-tracking resource for natural sentence reading. *Scientific Data*, 5(1):1–13.

Nora Hollenstein, Marius Tröndle, Martyna Plomecka, Samuel Kiegeland, Yilmazcan Özyurt, Lena A. Jäger, and Nicolas Langer. 2023. The ZuCo benchmark on cross-subject reading task classification with EEG

- and eye-tracking data. *Frontiers in Psychology*, 13:1028824.
- Zhicheng Huang, Xiaojie Jin, Chengze Lu, Qibin Hou, Ming-Ming Cheng, Dongmei Fu, Xiaohui Shen, and Jiashi Feng. 2023. Contrastive masked autoencoders are stronger vision learners. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Ashwin Kamble, Pradnya Ghare, Vinay Kumar, Ashwin Kothari, and Avinash Keskar. 2023. Spectral analysis of eeg signals for automatic imagined speech recognition. *IEEE Transactions on Instrumentation and Measurement*.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.
- Rui Li, Yiting Wang, Wei-Long Zheng, and Bao-Liang Lu. 2022a. A multi-view spectral-spatial-temporal masked autoencoder for decoding emotions with self-supervised learning. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 6–14.
- Shimin Li, Hang Yan, and Xipeng Qiu. 2022b. Contrast and generation make bart a good dialogue emotion recognizer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11002–11010.
- Yuanze Lin, Chen Wei, Huiyu Wang, Alan Yuille, and Cihang Xie. 2023. Smaug: Sparse masked autoencoder for efficient video-language pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2459–2469.
- Yue Ma, Tianyu Yang, Yin Shan, and Xiu Li. 2022. Simvtp: Simple video text pre-training with masked autoencoders. *arXiv preprint arXiv:2212.03490*.
- Nicolás Nieto, Victoria Peterson, Hugo Leonardo Rufiner, Juan Esteban Kamienkowski, and Ruben Spies. 2022. Thinking out loud, an open-access eeg-based bci dataset for inner speech recognition. *Scientific Data*, 9(1):52.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR.
- Jan-Mathijs Schoffelen, Robert Oostenveld, Nietzsche HL Lam, Julia Uddén, Annika Hultén, and Peter Hagoort. 2019. A 204-subject multimodal neuroimaging dataset to study language processing. *Scientific Data*, 6(1):17.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, 30.
- Yijun Wang, Xiaogang Chen, Xiaorong Gao, and Shangkai Gao. 2016. A benchmark dataset for ssvep-based brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1746–1752.
- Zhenhailong Wang and Heng Ji. 2022. Open vocabulary electroencephalography-to-text decoding and zero-shot sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 5350–5358.
- Nuwa Xi, Sendong Zhao, Haochun Wang, Chi Liu, Bing Qin, and Ting Liu. 2023. Unicorn: Unified cognitive signal reconstruction bridging cognitive signals and human language. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13277–13291.
- Yiqian Yang, Yiqun Duan, Qiang Zhang, Renjing Xu, and Hui Xiong. 2024. Decode neural signal as speech. *arXiv preprint arXiv:2403.01748*.
- Ke Yi, Yansen Wang, Kan Ren, and Dongsheng Li. 2024. Learning topology-agnostic eeg representations with geometry-aware modeling. *Advances in Neural Information Processing Systems*, 36.
- Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023a. A survey of large language models. *arXiv preprint arXiv:2303.18223*.
- Zijia Zhao, Longteng Guo, Xingjian He, Shuai Shao, Zehuan Yuan, and Jing Liu. 2023b. Mamo: Fine-grained vision-language representations learning with masked multimodal modeling. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1528–1538.

A Natural Masking Ratio of Datasets

To provide a clear perspective, we present the detailed statistics of the NMR of EEG feature sequences for three categories of reading task combinations in Table 7.

B Datasets and Implementation Details

We utilize the combination of both ZuCo v1.0 and ZuCo v2.0 to form the final ZuCo benchmark. The EEG features are collected with a 128-channel system under the sampling rate of 500Hz. After the noise canceling process, only 105 channels are used. There are 8 frequency bands determined in the ZuCo dataset as follows: theta1 (4–6 Hz), theta2 (6.5–8 Hz) alpha1 (8.5–10 Hz), alpha2 (10.5–13 Hz), beta1 (13.5–18 Hz) beta2 (18.5–30 Hz) and gamma1 (30.5–40 Hz) and gamma2 (40–49.5 Hz). The Hilbert transform is applied in each of these time series. The final features of the EEG are formed by concatenating features from all 8 frequency bands, resulting in a vector with a dimension of 840. For three reading tasks, we pre-train and fine-tune the models on “SR v1.0 + NR v1.0 + NR v2.0”. For four reading tasks, we choose the combination of “SR v1.0 + NR v1.0 + NR v2.0 + TSR v1.0”. For five reading tasks, the models are pre-trained and fine-tuned on “SR v1.0 + NR v1.0 + NR v2.0 + TSR v1.0 + TSR v2.0”. During pre-training, the datasets were split into training and testing sets in a 90% to 10% ratio. During the EEG-to-Text fine-tuning phase, the datasets were further divided into training, validation, and testing sets with an 80%, 10%, and 10% split respectively. The test set samples remained consistent throughout the above two stages. The dataset statistics of EEG-to-Text decoding are detailed in Table 8. Our training hyper-parameters are listed in Table 9. To ensure a fair comparison, we conducted both pre-training and fine-tuning for the EEG-to-Text decoding task using datasets with the same combinations of reading tasks.

C Generated Samples

We show more details in EEG-to-Text translation results generated on our models in Table 10, Table 11, and Table 12. In our experiments, we aim to select the same sentences from the test sets of three, four, and five reading tasks where feasible. This enables us to directly observe and compare the generated results with the ground truth across different task conditions.

D Subject-independent Performance

As reported in Table 1, we present the average BLEU-N and ROUGE-1 scores for all 30 subjects. However, considering the individual variations of brain activities during semantic processing and cognitive operations within different subjects, we further provide individual BLEU-N and ROUGE-1 scores for each subject. We use radar charts shown in Figure 3 and Figure 4 to visually represent these differences, allowing for an intuitive comparison across subjects. For a detailed numeric breakdown of these variances, refer to Table 13 and Table 14.

E Impact of the Masking Strategy

The masking strategy is crucial in Masked Autoencoders. For the text, the BERT masking strategy has proven highly effective. For the EEG modality, we introduce a pivotal design that involves mandatory masking of sentence-level EEG feature sequences, as detailed in Section 3.2. We delve into the impact of this strategy on the EEG-to-Text decoding task. Comparative results between random and forced masking strategies are presented in Table 15. The forced masking strategy outperforms the random masking strategy in the EEG-to-Text decoding, highlighting the efficacy of our proposed strategy in compelling the model to reconstruct the contextual semantics within sentence-level EEG feature sequences comprehensively.

F Impact of the Multi-Stream Design

Our investigation, as detailed in Table 16, reveals the transfer performance of a multi-stream design in the CET-MAE and E2T-PTR frameworks. The multi-stream approach, which provides the specialized handling of text and EEG using separate streams, outperformed a single joint stream design. Notably, in the E2T-PTR framework, leveraging the EEG-specific stream for fine-tuning yielded a marked improvement in EEG-to-Text task performance over a joint modality stream. This modality-focused approach appears to capitalize on the nuanced semantic information inherent in EEG embeddings, resulting in a more sophisticated and contextually relevant latent space. This is substantiated by the observed uptick in BLEU and ROUGE metrics. Our study underscores the criticality of fine-grained, modality-specific processing approaches in the domain of EEG-Text representation learning.

Reading Tasks	Missing Paris	Total word tokens	NMR(%)
SR v1.0 + NR v1.0+NR v2.0	90362	277966	32.51
SR v1.0 + NR v1.0+NR v2.0+TSR v1.0	137460	373817	36.77
SR v1.0 + NR v1.0+NR v2.0+TSR v1.0+TSR v2.0	204089	515979	39.55

Table 7: Statistics for natural masking ratios under three, four, and five reading tasks in ZuCo benchmarks.

Reading Task	Training Sample	Validation Sample	Testing Sample
SR v1.0 + NR v1.0+NR v2.0	10710	1332	1407
SRv1.0+NRv1.0+NRv2.0+TSRv1.0	14407	1790	1799
SRv1.0+NRv1.0+NRv2.0+TSRv1.0+TSRv2.0	18791	2287	2404

Table 8: Dataset Statistics of the EEG-to-Text decoding. SR: Normal Reading (Sentiment), NR: Normal Reading (Wikipedia), TSR: Task Specific Reading (Wikipedia).

Hyperparameters	Pre-training			Fine-tuning		
Models		CET-MAE		E2T-PTR		
Reading Tasks	3	4	5	3	4	5
Datasets Splits		9:1			8:1:1	
Epochs		100		50	40	40
Batch Size		32			32	
Learning Rate		5e-7		2e-7	2e-5	2e-5
Optimizer		AdamW, weight decay= 1e-2, betas =(0.9,0.999)				
LR Scheduler		Cosine Annealing, T_max=20				
GPUs		RTX4090				

Table 9: Implementation details in our pre-training and fine-tuning.

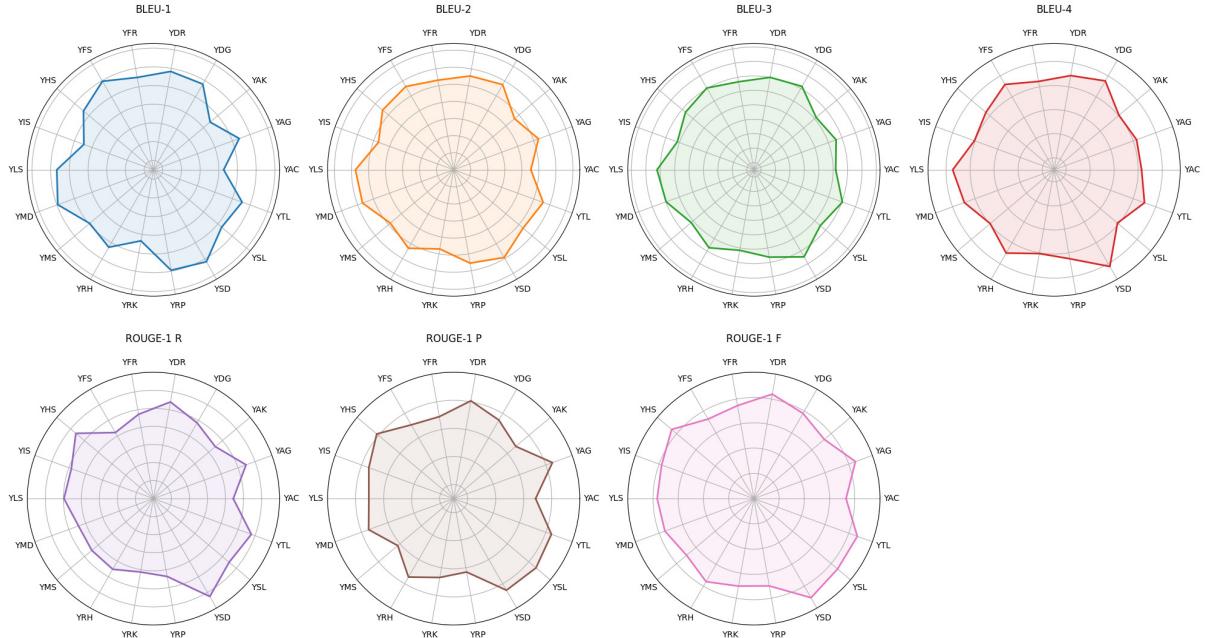


Figure 3: The radar chart of 18 subjects from Subject YAG to YSD on each metric.

	Ground Truth: At the urging of his wife , Columba, a devout Mexican Catholic , the Protestant Bush became a Roman Catholic.
(1)	E2T-PTR: the time of his wife , hea, he former Catholic Catholic , he actor pastorman a Catholic Catholic .
(2)	Ground Truth: While attending a motorcycle race, he met a local girl named Columba Garnica Gallo, <i>whom he</i> eventually married . E2T-PTR: in the local school, he was his man boy named Marya.ett,o, <i>who he</i> later married .
(3)	Ground Truth: He then enrolled at Phillips Andover, a private boarding school in Massachusetts already attended by his brother George. E2T-PTR: was went in the Academy Mary College where private school school in Massachusetts . known by his father .
(4)	Ground Truth: He took a job in real estate with Armando Codina, a 32-year-old Cuban immigrant and self-made American millionaire . E2T-PTR: was a job as the estate in theando lice in who company-year-old Italian immigrant . former-made millionaire millionaire .
(5)	Ground Truth: After earning his degree , Bush went to work in an entry level position in the international division of Texas Commerce Bank, which was run by Ben Love . E2T-PTR: the his bachelor in he became to work for the office- position at the Department banking of the Instruments.. where was later by theitott
(6)	Ground Truth: He later became <i>an educator</i> , teaching music theory at the University of the District of Columbia ; he was also director of the District of Columbia Music Center jazz workshop band. E2T-PTR: was became <i>a American</i> and and at and and the University of California West of Columbia . and also also a of the school of Columbia's Department. department..
(7)	Ground Truth: Bush stayed in Houston with another family to finish the school year , and spent most summers and holidays at the family estate, known as the Bush Compound . E2T-PTR: was in the until his family, raise his year year . and then the of in summers there the family's . including as the Bush Ranchound .
(8)	Ground Truth: Robert Henry Dee (born May 18, 1933 in Quincy, Massachusetts) is a former three-sport letterman at Holy Cross College who was one of the first <i>players</i> signed by the Boston Patriots in 1960. E2T-PTR: Frost, (born April 5, 18) New, Massachusetts) is a retired United-timeport star carrier and the Cross College . <i>played a of the founders African to by the</i> University Celtics. the.

Table 10: EEG-to-Text decoding example results on test sentences under three reading tasks. **Bold** words indicate exact match, *Italic* words indicate semantic resemblance, and Underline words indicate error match.

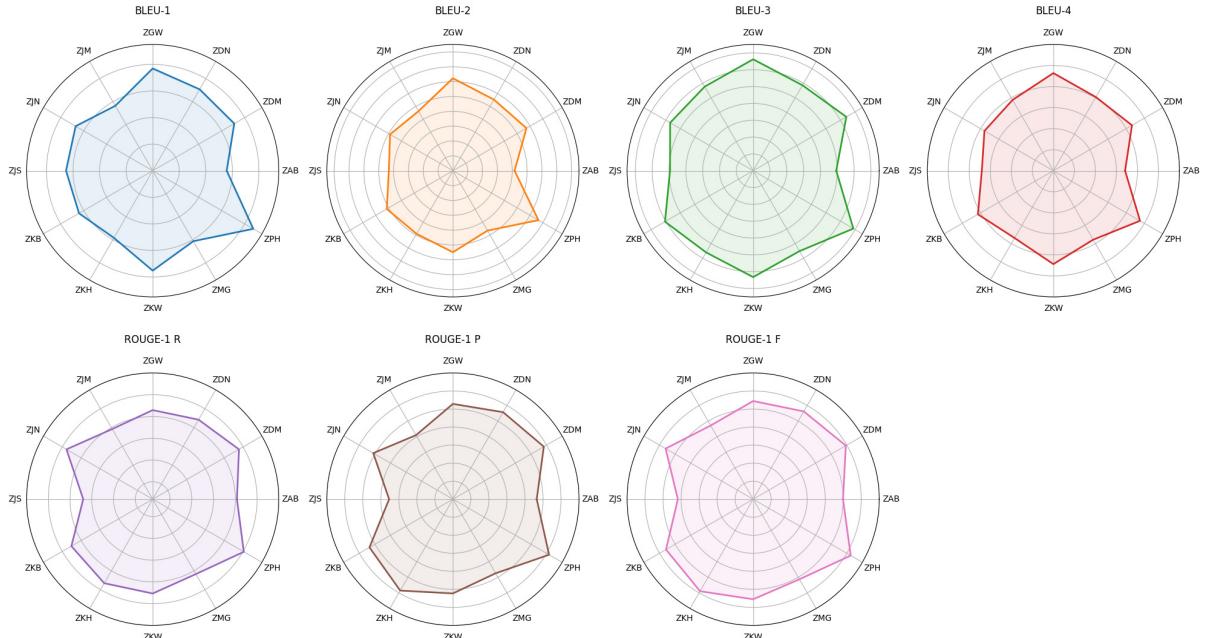


Figure 4: The radar chart of 12 subjects from Subject ZKW-ZJS on each metric.

(1)	Ground Truth: At the urging of his wife , Columba, a devout Mexican Catholic , the Protestant Bush became a Roman Catholic . E2T-PTR: the academy of his mother , hea, she young Catholic , she young preacher co an Catholic Catholic in
(2)	Ground Truth: While attending a motorcycle race, he met a local girl named Columba Garnica Gallo , whom he eventually married . E2T-PTR: serving the Louisiana school he he met a man hero named Dela Jacksonett.iennne , who he would struck.
(3)	Ground Truth: He then enrolled at Phillips Andover, a private boarding school in Massachusetts already attended by his brother George . E2T-PTR: was returned in the University Mary College Massachusetts public school school in the. owned by his father .
(4)	Ground Truth: He took a job in real estate with Armando Codina , a 32-year-old Cuban immigrant and self-made American millionaire . E2T-PTR: was many second as the estate with theando Feric, where local-year-old hotel shipping who hotel-trained millionaire millionaire who
(5)	Ground Truth: After earning his degree, Bush went to work in an entry level position in the international division of Texas Commerce Bank, which was run by Ben Love . E2T-PTR: a his Ph at he went to work for the apprentice- role at the Springfield trade of the Instruments. at working he subsequently by Jamesoittt
(6)	Ground Truth: He later became an educator, teaching music theory at the University of the District of Columbia ; he was also director of the District of Columbia Music Center jazz workshop band. E2T-PTR: was earned president assistant at and English at at the University of Wisconsin Arts of Columbia , and also the a of the Special School Columbia Library Project . line..
(7)	Ground Truth: Bush stayed in Houston with another family to finish the school year , and spent most summers and holidays at the family estate , known as the Bush Compound . E2T-PTR: was in Hollywood for his oil, work his term year , and to the summers and holidays at the sprawling estate , the as the Bush Compound .
(8)	Ground Truth: Robert Henry Dee (born May 18, 1933 in Quincy, Massachusetts) is a former three-sport letterman at Holy Cross College who was one of the first players signed by the Boston Patriots in 1960. E2T-PTR: Joseph Bol,born July 22, 1923 Ball, Massachusetts) is best former Republican-timeides quarterbackman who the Cross College , is elected of the founder " to to the University Bruins. 1993.

Table 11: EEG-to-Text decoding example results on test sentences under four reading tasks. **Bold** words indicate exact match, *Italic* words indicate semantic resemblance, and Underline words indicate error match.

(1)	Ground Truth: At the urging of his wife , Columba, a devout Mexican Catholic , the Protestant Bush became a Roman Catholic . E2T-PTR: the academy of his mother , hea, she young Catholic , she young and accepted a Catholic Catholic in
(2)	Ground Truth: <u>While attending a motorcycle race</u> , he met a local girl named Columba Garnica Gallo , whom he eventually married. E2T-PTR: <u>serving a motorcycle race</u> , he met a local girl named Columba Garnica Gallo , whom he eventually married.
(3)	Ground Truth: <u>He then enrolled at Phillips Andover</u> , a private boarding school in Massachusetts already attended by his brother George . E2T-PTR: <u>was enrolled at Phillips Andover</u> , a private boarding school in Massachusetts already attended by his brother George .
(4)	Ground Truth: He took a job in real estate with Armando Codina , a 32-year-old Cuban immigrant and self-made American millionaire . E2T-PTR: was his job with the estate with theco Ferela and and firm-year-old firm shipping who hotel-trained millionaire merchant .
(5)	Ground Truth: After earning his degree, Bush went to work in an entry level position in the international division of Texas Commerce Bank, which was run by Ben Love . E2T-PTR: a his degree , Bush went to work in an entry level position in the international division of Texas Commerce Bank, which was run by Ben Love .
(6)	Ground Truth: <u>He later became an educator</u> , teaching music theory at the University of the District of Columbia ; he was also director of the District of Columbia Music Center jazz workshop band. E2T-PTR: was became president educator , teaching music theory at the University of the District of Columbia ; he was also director of the District of Columbia Music Center jazz workshop band.
(7)	Ground Truth: Bush stayed in Houston with another family to finish the school year , and spent most summers and holidays at the family estate , known as the Bush Compound . E2T-PTR: is in Hollywood for his company, work his war year . and <u>enrolled</u> the summers and holidays at the sprawling estate , the as the Bush Compound .
(8)	Ground Truth: Robert Henry Dee (born May 18, 1933 in Quincy, Massachusetts) is a former three-sport letterman at Holy Cross College who was one of the first players signed by the Boston Patriots in 1960. E2T-PTR: Emerson Dee (born May 18, 1933 in Quincy, Massachusetts) is a former three-sport letterman at Holy Cross College who was one of the first players signed by the Boston Patriots in 1960.

Table 12: EEG-to-Text decoding example results on test sentences under five reading tasks. **Bold** words indicate exact match, *Italic* words indicate semantic resemblance, and Underline words indicate error match.

Subjects	YAG	YAK	YMS	YHS	YSL	YRK	YRH	YDR	YIS	YRP	YLS	YTL	YFR	YDG	YAC	YFS	YMD	YSD
BLEU-1	46.23	46.67	45.65	46.12	46.50	46.34	45.90	46.13	45.90	46.45	46.12	46.56	44.75	46.78	46.28	46.51	46.89	45.65
BLEU-2	28.98	28.93	28.80	28.94	29.57	29.10	28.88	29.28	28.78	29.41	28.94	29.63	27.79	29.60	28.70	29.93	29.82	28.52
BLEU-3	18.07	17.74	17.69	17.85	18.32	17.70	17.82	18.76	17.57	18.45	17.64	18.44	16.90	18.22	17.44	18.87	18.52	17.88
BLEU-4	11.27	10.85	11.09	11.04	11.22	10.70	10.89	12.10	10.64	11.82	10.67	11.44	9.88	11.34	10.50	12.09	11.48	11.18
ROUGE1-R	35.21	35.66	35.73	34.99	36.00	35.23	35.86	35.17	34.77	35.37	35.13	35.58	34.24	34.86	35.30	35.62	35.94	35.03
ROUGE1-P	41.55	42.46	42.88	41.91	43.32	41.69	42.36	41.53	41.29	42.20	42.20	42.04	40.27	41.37	42.30	42.84	42.97	41.90
ROUGE1-F1	38.02	38.65	38.87	38.04	39.22	38.09	38.73	37.97	37.66	38.39	38.24	38.45	36.92	37.72	38.41	38.79	39.04	38.05

Table 13: Subject-independent Performance of BLEU-N(%) and ROUGE-1 from Subject YAG to YSD.

Subjects	ZKW	ZPH	ZAB	ZKB	ZMG	ZJN	ZDN	ZJM	ZGW	ZDM	ZKH	ZJS
BLEU-1	37.99	38.49	38.16	38.02	37.97	38.31	37.84	38.05	38.36	38.15	38.19	37.11
BLEU-2	20.83	21.07	20.83	20.89	21.14	20.74	20.81	20.73	21.58	20.92	21.00	20.34
BLEU-3	10.82	11.19	11.14	10.91	11.40	11.16	11.19	10.72	11.75	10.90	11.13	10.48
BLEU-4	5.76	6.01	6.18	5.70	6.34	6.18	6.27	5.55	6.60	5.82	6.29	5.49
ROUGE1-R	25.34	25.21	24.51	25.38	25.44	25.53	25.46	25.27	26.15	25.08	25.78	24.15
ROUGE1-P	30.44	30.43	29.39	30.74	30.55	30.48	30.31	30.27	31.14	30.10	31.02	28.84
ROUGE1-F1	27.55	27.45	26.62	27.67	27.64	27.65	27.53	27.43	28.30	27.24	28.04	26.17

Table 14: Subject-independent performance of BLEU-N(%) and ROUGE-1 from Subject ZKW to ZJS.

Method	Training Sample	Mask Stragety	BLEU-N(%)				ROUGE-1 (%)		
			N=1	N=2	N=3	N=4	P	R	F
E2T-PTR	10710	Random Mask	40.51	24.10	14.05	8.24	35.38	29.68	32.17
	10710	Force Mask	42.09	25.13	14.84	8.99	35.86	30.01	35.61

Table 15: Investigating the impact of mask strategy in EEG feature sequences during CET-MAE pre-training.

CET-MAE	E2T-PTR	Model	Training Sample	BLEU-N(%)				ROUGE-1(%)		
				N=1	N=2	N=3	N=4	P	R	F
X	Joint Stream	Joint Stream	10710	41.60	24.53	14.19	8.35	35.34	29.57	32.09
✓	Joint Stream	Joint Stream	10710	41.61	24.57	14.34	8.52	35.74	29.79	32.37
✓	EEG Stream	EEG Stream	10710	42.09	25.13	14.84	8.99	35.86	30.01	32.61

Table 16: We validated the performance impact of multi-stream design on pre-training and downstream tasks. The ✓ indicates the use of a multi-stream design during pre-training, while the X indicates no use.

Foundational GPT Model for MEG

Richard Csaky^{*,1,2,5}, Mats W.J. van Es^{1,2}, Oiwi Parker Jones^{2,3,4}, and Mark Woolrich^{1,2}

¹Oxford Centre for Human Brain Activity, Department of Psychiatry, University of Oxford, OX3 7JX, Oxford, UK

²Wellcome Centre for Integrative Neuroimaging, OX3 9DU, Oxford, UK

³Department of Engineering Science, University of Oxford, OX1 3PJ, Oxford, UK

⁴Jesus College, OX1 3DW, Oxford, UK

⁵Christ Church, OX1 1DP, Oxford, UK

{richard.csaky@psych, mats.vanes@psych,
oiwi.parkerjones@eng, mark.woolrich@ohba}.ox.ac.uk

Abstract

Deep learning techniques can be used to first training unsupervised models on large amounts of unlabelled data, before fine-tuning the models on specific tasks. This approach has seen massive success for various kinds of data, e.g. images, language, audio, and holds the promise of improving performance in various downstream tasks (e.g. encoding or decoding brain data). However, there has been limited progress taking this approach for modelling brain signals, such as Magneto-/electroencephalography (M/EEG). Here we propose two classes of deep learning foundational models that can be trained using forecasting of unlabelled MEG. First, we consider a modified Wavenet; and second, we consider a modified Transformer-based (GPT2) model. The modified GPT2 includes a novel application of tokenisation and embedding methods, allowing a model developed initially for the discrete domain of language to be applied to continuous multichannel time series data. We also extend the forecasting framework to include condition labels as inputs, enabling better modelling (encoding) of task data. We compare the performance of these deep learning models with standard linear autoregressive (AR) modelling on MEG data. This shows that GPT2-based models provide better modelling capabilities than Wavenet and linear AR models, by better reproducing the temporal, spatial and spectral characteristics of real data and evoked activity in task data. We show how the GPT2 model scales well to multiple subjects, while adapting its model to each subject through subject embedding. Finally, we show how such a model can be useful in downstream decoding tasks through data simulation. All code is available on GitHub¹.

1 Introduction

Unsupervised learning provides a dataset-agnostic method for learning shared representations. Unsupervised learning techniques can be further differentiated between those that aim to learn interpretable representations and those that are purely data-driven. In functional neuroimaging, interpretable models can provide neuroscientific insights that are especially useful for rest or spontaneous data, where there is no known external stimuli or behaviour

^{*}Corresponding author.

¹<https://github.com/ricsinaruto/MEG-transfer-decoding>

associated with the brain activity (Baker et al., 2014). Models designed without focusing on interpretability can be used to generalise over multiple heterogeneous datasets and provide a pretrained foundation model (Yuan et al., 2024). Such foundation models can then be fine-tuned for downstream tasks on smaller amounts of data for which relevant labels are available (e.g. to do encoding or decoding). By leveraging large amounts of data, the hope is that the foundation model will be capable of generalising to new data types and provide improvement over just training the model on the labelled dataset. This is especially useful for brain-computer interface (BCI) applications.

The concept of using vast amounts of data to boost performance in downstream tasks originates from deep learning. Perhaps the most successful recent example is that of large language models, trained on diverse data sources and demonstrating enhanced capabilities over task-specific models in a multitude of language-related tasks (e.g., translation, summarisation) (Brown et al., 2020). This can also be viewed as a form of transfer learning. Zero-shot performance is obtained when no fine-tuning is done for the downstream task. Several factors enabled the success of large language models, including data scale, model size, fast GPUs, and effective neural network architectures (Kaplan et al., 2020; Fedus et al., 2022; Sutton, 2019). To adopt this paradigm for electrophysiology data, the primary obstacles are the model architecture and data size. In this paper, we focus on the former.

We aim to design general models well-suited to multichannel timeseries that can scale effectively. We also focus on foundational models that can be trained using forecasting (as opposed to masked prediction), which are causal and can generate data recursively. This allows for the interrogation of learned spatio-temporal dynamics over long temporal horizons. Self-supervised learning (SSL) has emerged as a promising approach for learning useful representations from unlabelled electrophysiological data. SSL reformulates an unsupervised learning problem as a supervised one by exploiting inherent structure in the data to generate "pseudo-labels". Forecasting is one such SLL task, as is reconstruction of masked sequence segments (Wang et al., 2023). In the context of electrophysiology, recent works have proposed SSL tasks tailored to the temporal and multivariate nature of neural time series data (Gohil et al., 2022; Banville et al., 2021; Kostas et al., 2021; Wang et al., 2023; Cui et al., 2023).

In our quest for designing expressive foundational models of MEG data, we can look to artificial intelligence domains with similar characteristics, such as audio or natural language processing. These domains share some similarities with MEG data, like the sequential nature of the modality. However, while audio data is also a continuous timeseries, it only contains a single channel and comes at a much higher sampling rate compared to M/EEG data. Language data is perhaps even more different as its timeseries are comprised of distinct units (words) from a finite vocabulary set. As such, starting with models developed for these domains and adapting them to handle the nuances of M/EEG data is a promising approach. In this work, we adapt two such approaches. First, we adapt Wavenet, originally developed for forecasting audio data (van den Oord et al., 2016); and second, we adapt a Transformer architecture (Vaswani et al., 2017) in the form of GPT2, originally developed for forecasting language (Radford et al., 2019).

Wavenet has been shown to be an effective model for forecasting time series, through its dilated convolutional architecture, which is fast and parameter-efficient (van den Oord et al., 2016). Here, we modify Wavenet to handle multichannel MEG data by first doing a channel-dependent quantisation (WavenetFullChannel) and then introducing a mixing layer

across the channel dimension (`WavenetFullChannelMix`).

In recent years, the Transformer architecture has driven a second deep learning revolution (Vaswani et al., 2017). Transformers use attention to model complex dependencies in long sequences, providing a more flexible inductive bias well-suited to language modelling and other tasks involving highly structured sequential data (Devlin et al., 2019; Brown et al., 2020). For example, self-attention provides direct connectivity between any two time steps, capturing long-range dependencies. The parallelisable architecture allows for more efficient computation compared to recurrent models. Transformers have also been applied to time series data with promising results (Wen et al., 2022). Zhou et al. (2021) adapted the self-attention mechanism for long-range forecasting and demonstrated state-of-the-art performance on multiple public datasets.

Here, we modify the Transformer architecture used in GPT2 (Radford et al., 2019) to handle multichannel MEG data. Specifically, we do a channel-independent quantisation, but augment the input with channel-embeddings to signify to the model which timeseries corresponds to which channel. We refer to this model as `ChannelGPT2`. While previous work has employed various modifications to the Transformer architecture we posit that keeping the categorical nature of the sequences and doing next time-step forecasting are essential for effective modelling. Our methods are further detailed in Section 3.

Both the modified Wavenet and `ChannelGPT2` models also include subject embeddings (Csaky et al., 2023b) and task condition embeddings. The latter provides the model with information about external stimuli synchronised to the MEG time-course. This means that the model can be treated as both a forecasting and an encoding model.

We evaluate how well the modified Wavenet and `ChannelGPT2` models can perform as foundational models. First, we assess how well the fitted models can recursively generate data with the same spatial, temporal and spectral characteristics as real MEG data. We evaluate this both at the subject- and group-level. Second, we see how well the foundation models can be used on downstream tasks, through data simulation. Finally, we show through several ablation studies how the best performing model, `ChannelGPT2`, leverages condition and channel embeddings.

2 Results

2.1 Next time-step prediction does not capture performance

We wanted to evaluate how well the modified Wavenet and `ChannelGPT2` models can perform as foundational models. One way to assess this is through the forecasting performance, i.e. the prediction accuracy of the label at the next time point. Here, we evaluated the performance of our two modified versions of Wavenet (`WavenetFullChannel` and `WavenetFullChannelMix`) alongside `ChannelGPT2`, as summarised in Table 1. For comparison, we also evaluated the performance of a linear autoregressive (AR) model of order 255. All deep learning models were trained on tokenised and embedded inputs, and cross-entropy loss was used to predict categorical probability distributions over tokens. AR models were trained on the continuous data using the mean-squared error loss, and we simply binned the predicted continuous output to compute accuracy and compare with other models.

The next-timestep forecasting accuracy for different models on a sample subject is shown in

Model	Univariate	Tokenised	Linear
AR(255)	yes	no	yes
WavenetFullChannel	yes	yes	no
WavenetFullChannelMix	no	yes	no
ChannelGPT2	yes	yes	no

Table 1: The 4 main models presented in the results section. Univariate means that the channel dimension is treated as a batch dimension during training. Non-tokenised models predict continuous values directly instead of categorical distributions.

Model	MSE	Top-1 Accuracy	Top-5 Accuracy
Repeat baseline	0.024	1.5	7.6
AR(255)	0.016	1.5	7.5
WavenetFullChannel	0.026	2.0	9.8
WavenetFullChannelMix	0.022	2.2	10.8
ChannelGPT2	0.023	2.2	10.9

Table 2: Test data next-timestep prediction performance across various models. Accuracy values are given in percentages. Chance-level is 1/256, however predicting the majority class (quantised value) is somewhat higher, since the distribution over bins is not uniform.

Tabel 2. Beyond standard accuracy (the number of true positives divided by the number of all examples), we also evaluated top-5 accuracy, counting a prediction as correct if the true bin was within the 5 most probable bins. Surprisingly, all models performed only slightly better than a naive baseline of repeating the previous timestep’s value.

As expected, the linear AR model had lower MSE but worse accuracy than the nonlinear models. This can be because MSE measures the distance of the prediction to the target, while accuracy is only 1 if the prediction is in the target bin. Thus, it may be that the AR model always predicts values that are slightly closer to the target, but never quite falling in the target bin. While WavenetFullChannel appears to be worse, WavenetFullChannelMix and ChannelGPT2 have nearly identical performance.

The choice of sampling rate can affect forecasting performance. A higher sampling rate makes the task easier as consecutive timesteps are more correlated, however this might make the model focus on very short-range temporal dependencies and overfit to noise. We analysed forecasting performance in relation to sampling rate in Supplementary Section A.2.

2.2 PSD of generated data matches real data

We have seen how forecasting accuracy does not differentiate well between the candidate foundation models. Despite this, the models may perform differently when evaluated using other criteria. For example, a good foundational model should be expected to be able to recursively generate data that looks like the real data. Here, we first assess the models’ ability to do this using the power spectra.

For deep learning models we used top-p sampling with $p = 80\%$ (unless otherwise noted in the figure caption) to recursively generate data. We generated 3600 seconds with all models. For models that have task-conditioning (all except AR(255)) we use the task label timeseries

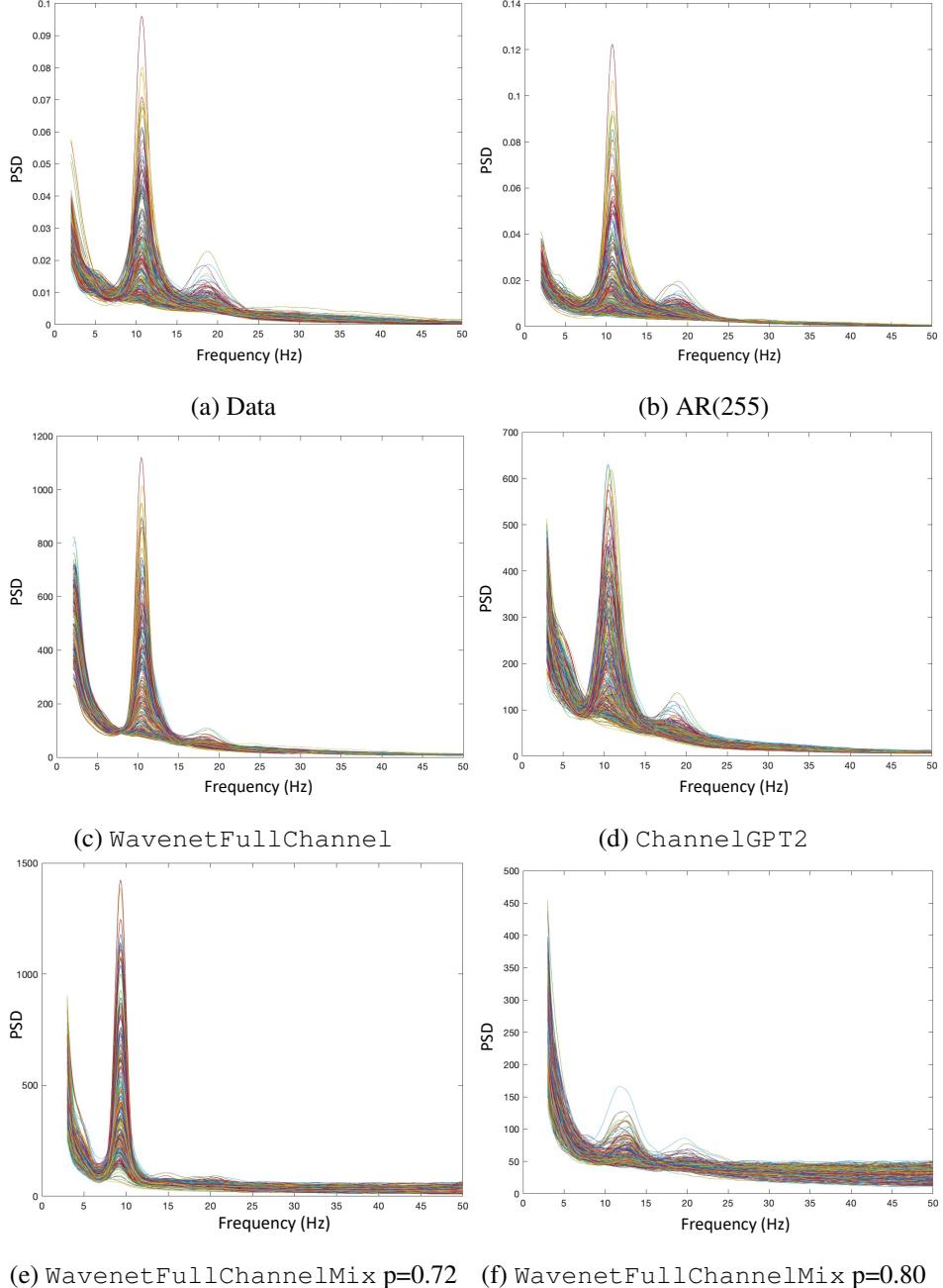


Figure 1: Comparison of generated data power spectral density (PSD) across models. Each line represents a different MEG channel. Note that in e) and f) the value of p corresponds to the probability mass used for top- p sampling.

from the training set. The models in this section were trained on a single sample subject.

Generated token sequences are first de-tokenised and then the power spectral density (PSD) is computed on the continuous data. Figure 1 compares the PSD of the generated data across our models. Qualitatively, it is clear that AR(255) reproduces PSDs that match best with those computed directly on the MEG data, while WavenetFullChannel and ChannelGPT2 are not far behind. All models capture the characteristic $1/f$ shape, and peaks at 10 and 19

Hz, likely related to alpha and beta band activity. Notably, `WavenetFullChannel` has reduced power at the 19 Hz peak, which could indicate issues in capturing higher frequency dynamics.

The performance of `WavenetFullChannelMix` was found to depend on the choice of the probability mass p , used in top- p sampling from the model to generate the data. p prohibits the generation of any tokens that are not within the top- p cumulative probability mass. Even slight modifications (e.g., $p = 0.72$ vs. $p = 0.8$ for `WavenetFullChannelMix`) result in large differences in the frequency of the two peaks in the PSDs computed on the generated data, and also the width of the peaks. Ultimately both top- p values provide subpar PSDs compared to channel-independent models. This is likely to be due to overfitting, as channel-mixing models lack the implicit regularisation of modelling each channel separately. The implicit regularisation is a consequence of having much more examples and much fewer input features when treating channels as a batch dimension.

2.3 HMM statistics of generated data match real data

Next, we assessed how well the fitted models can recursively generate data with the same spatial, temporal and spectral multi-channel characteristics as real MEG data. HMMs are an established way for doing unsupervised discovery of multi-channel dynamics in real neuroimaging data, and have been used to characterise the spatial, temporal and spectral characteristics of brain networks in MEG data (Rabiner, 1989; Vidaurre et al., 2018a).

Separately, we trained each model on a single sample subject (with condition embeddings), generated multi-channel data, and then inferred a 12-state HMM, with each state modelling the multi-channel data as a multi-variate Normal (MVN) distribution. The number of states was chosen based on previous work (Vidaurre et al., 2018a). Example state time-courses generated from all models are plotted in Figure 2 to qualitatively illustrate the differences in the generated dynamics. Note that since a separate HMM is trained each time, the states are not matched between models or with the real data. States could be matched post-HMM training by looking for similarity in the observation models, but here we opted to plot unmatched state time-courses. Since each generated timeseries is different, comparison across state timecourses would not be useful anyway.

We extracted four summary statistics on the inferred state timecourses and compared their distribution over states. These statistics are shown across models in Figure 3, alongside those for an HMM trained on the real multi-channel MEG data. Across the four summary statistics we can see that the real data has high variance in the distribution over states. AR(255) and `WavenetFullChannelMix` fail to produce data with variable state statistics, and even the mean over states is not captured well. `WavenetFullChannel` does a great job at capturing the mean of the state distributions, but still produces data with relatively invariant states. `ChannelGPT2` seems to best capture the distributions across all four statistics, especially for the mean interval and switching rate. This shows that Transformer-based models can generate data that better matches the HMM-inferred dynamics of real MEG data.

In addition to state statistics, we also computed the power spectra of each state across the timeseries. In MEG data different states might capture oscillatory activity with specific frequencies. The extracted power spectra from the inferred state time courses is shown in Figure 4. We can see that the HMM trained on the MEG data contains many states that capture the 10 Hz peak, with fewer states having a 20 Hz peak. It is also clear that the

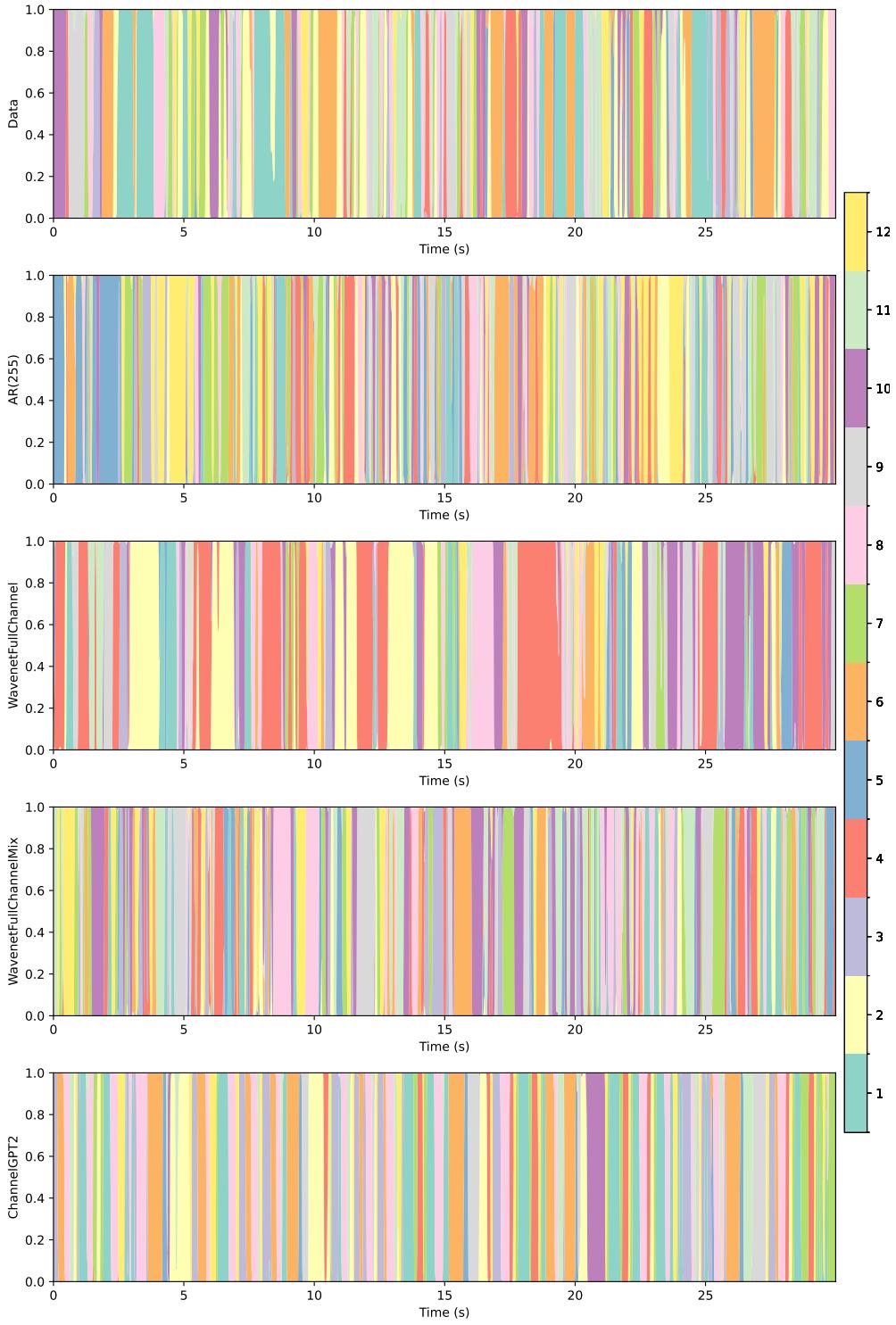


Figure 2: Example state timecourses from the HMMs trained on each model’s generated data (rows). Each state is represented by a different colour. Note that state indices and timecourses are not matched across models.

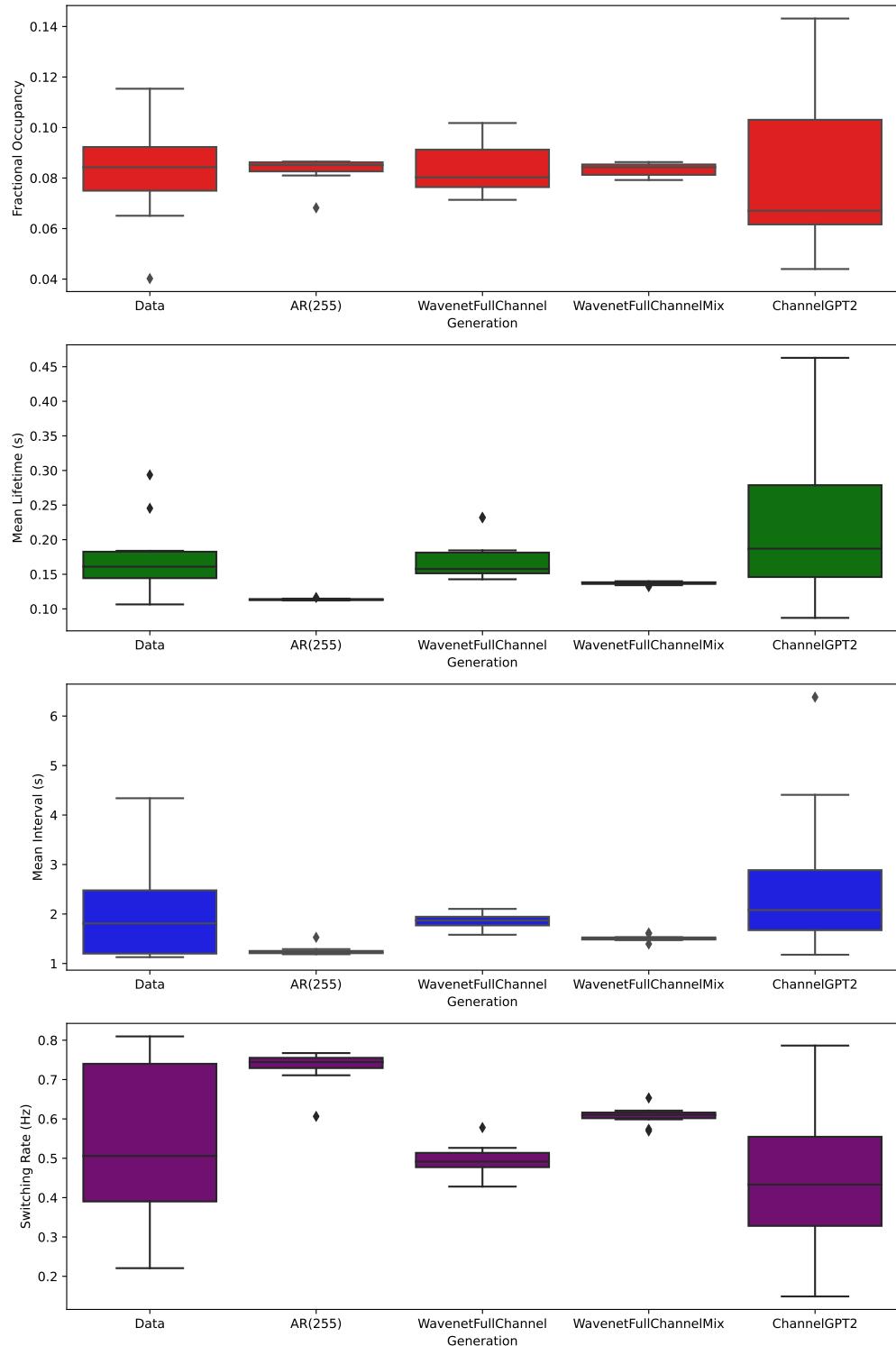


Figure 3: Distributions across the 12 states inferred by an HMM using multi-channel data generated from the different models, or using the real sensor space MEG data (columns). Four different summary statistics are shown describing the state dynamics (rows).

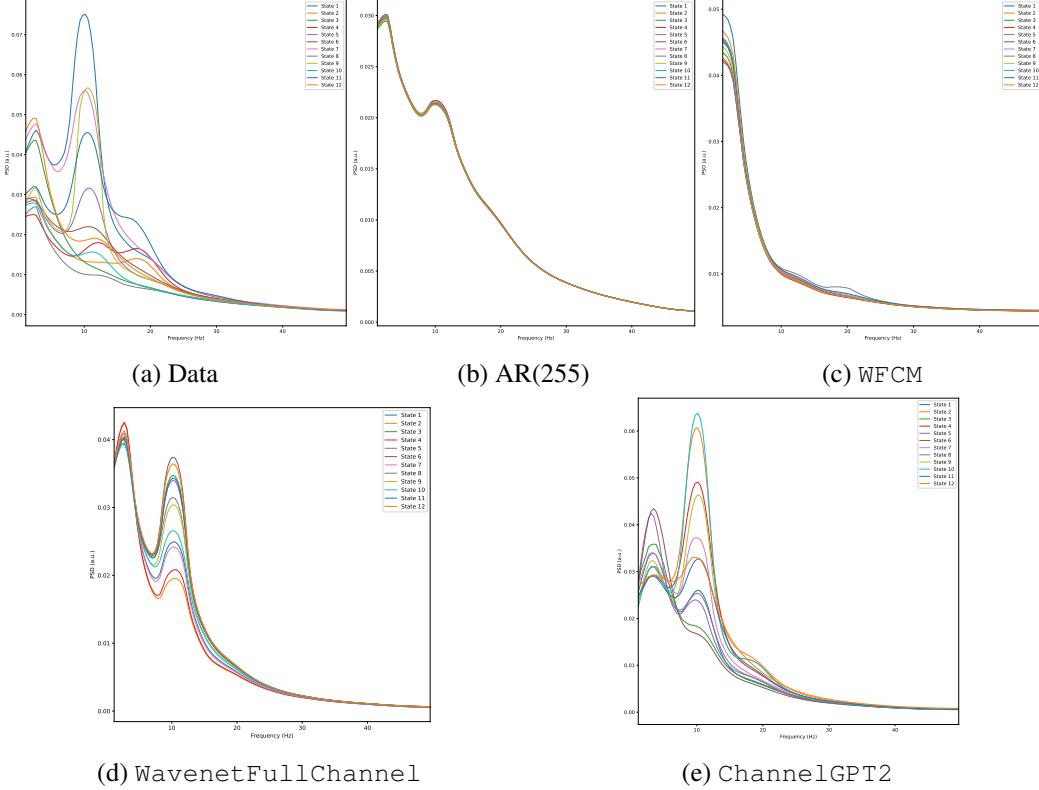


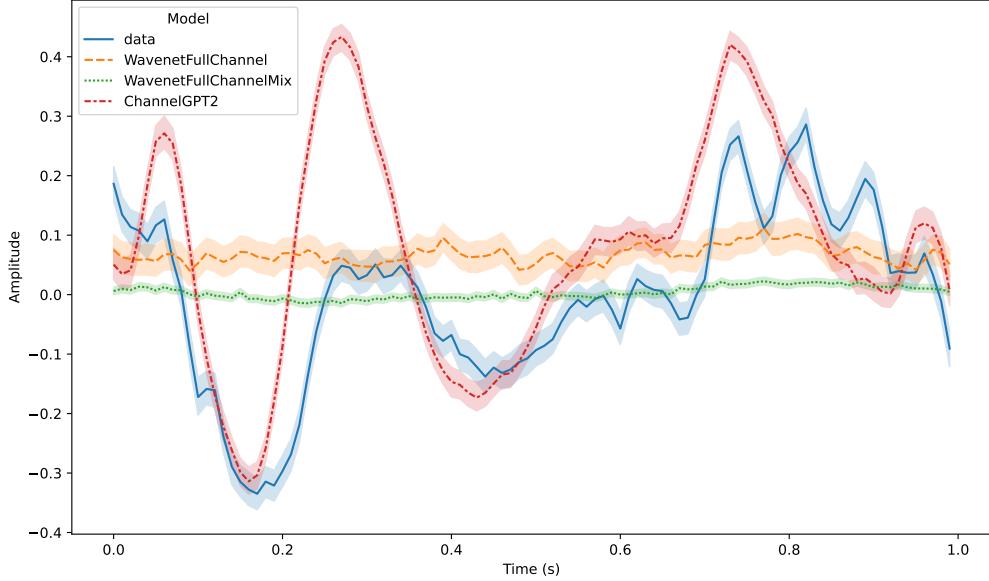
Figure 4: Power spectral density of HMM states inferred on the generated data of each model and on real MEG data. WFCM refers to WavenetFullChannelMix. Each line is the PSD of a different state. Note that states are not matched across models. Horizontal axis represents frequency in Hz.

states of the HMM fitted to the WavenetFullChannelMix generated timeseries do not contain these spectral peaks. While the AR(255) does contain states with a 10 Hz peak, the shape does not match the data well, and also states do not show the same variability as in real data.

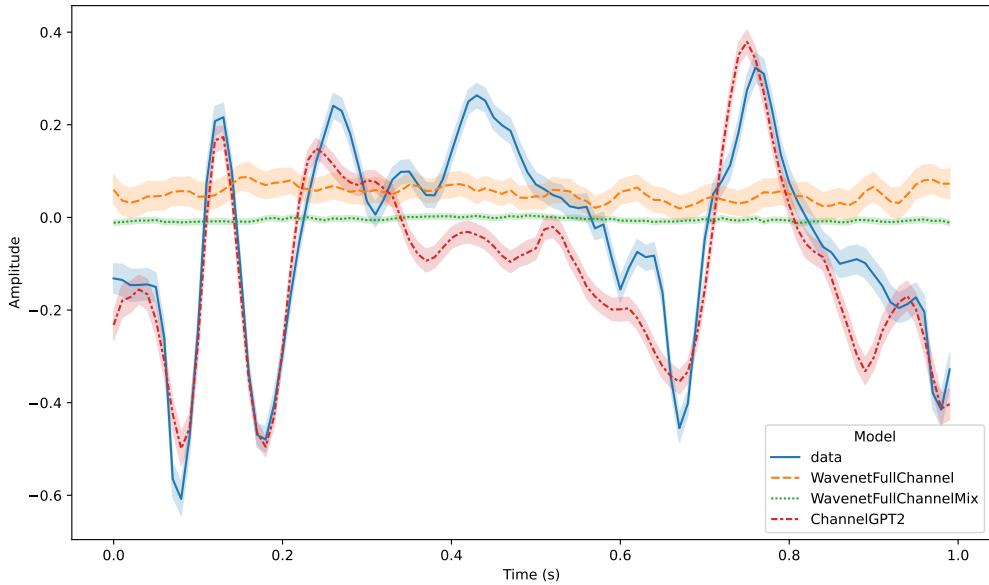
In contrast, ChannelGPT2, matches the state PSDs of the real data very well, further demonstrating the superiority of Transformer models in capturing complex neural dynamics. While WavenetFullChannel also improves substantially over the AR(255) power spectra, it falls short in capturing the 20 Hz peak and the heterogeneity between states observed in the real data and the generated data of ChannelGPT2. This and previous analyses show that the combination of channel-independence and a Transformer-based architecture are critical for matching the dynamics of real data.

2.4 Evoked activity in generated data matches real data

The analyses in the previous section considered metrics for assessing the ability of the candidate foundation models to generate timeseries without requiring any a priori knowledge of the timing of brain activity. Here, we use knowledge of the experimental task timings in the Cichy et al. (2016) data, to provide insight into the ability of the foundational models to generate realistic task data.



(a) Frontal channel (MEG0111)



(b) Visual channel (MEG2332)

Figure 5: Comparison of evoked timecourses of 2 channels across our task-conditioned models. Stimulus onset is at 0 seconds and offset is at 500ms. The peak occurring after 500 ms indicates a visual response to the offset of the stimulus (removal of the image). Shading indicates variability across trials.

The models in this section were trained on a single sample subject. As mentioned before, we used the task label timeseries from the training data when generating data with our models. If the models properly incorporate this conditioning, the generated data should reflect aligned task-related activity similar to real data.

By simple epoching of the generated timeseries based on the known task labels, we can

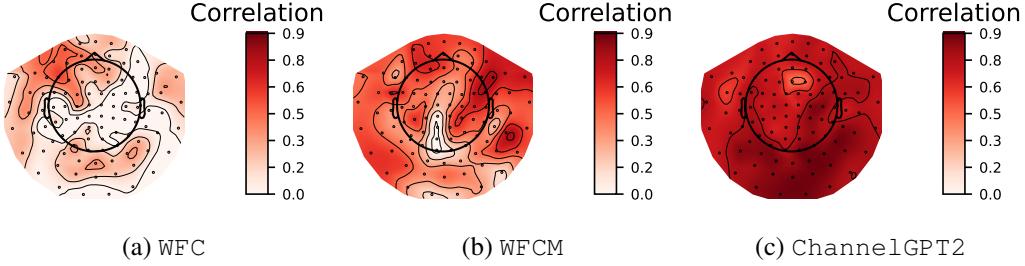


Figure 6: Correlation between the time-courses of the mean (over individual epochs) evoked responses from the real data and mean evoked responses generated by each model. The correlation values are visualised across sensors. WFC refers to WavenetFullChannel and WFMC refers to WavenetFullChannelMix. Darker reds indicate higher correlation.

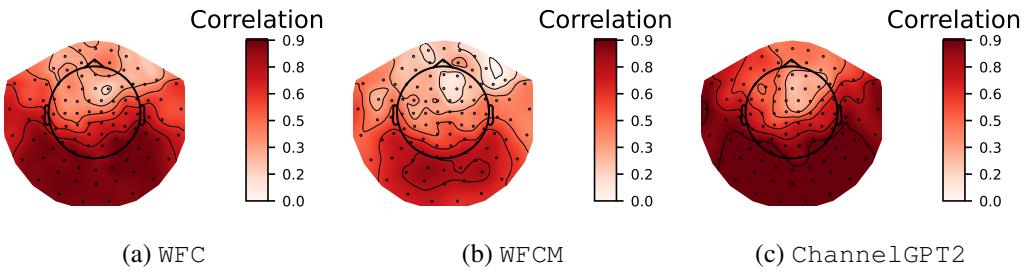


Figure 7: Correlation between the time-courses of the variance (over individual epochs) of the mean evoked responses from the real data and the variance of the mean evoked responses generated by each model. The correlation values are visualised across sensors. WFC refers to WavenetFullChannel and WFMC refers to WavenetFullChannelMix. Darker reds indicate higher correlation.

compute evoked responses in the data generated by our models. We do this for all models except AR(255) as it did not include task labels in its model. To compare the shape of average evoked responses, we averaged over all epochs in both real data and the generated timeseries. This results in data of shape $\bar{\mathbf{X}} \in \mathbb{R}^{C \times T}$ where $C = 306$ is the number of channels and $T = 1000$ ms is the trial/epoch length.

The evoked responses across our models and the real data in a frontal and a visual channel are shown in Figure 5. While Wavenet models completely fail to capture the evoked time-course, ChannelGPT2 does a remarkably good job, especially in the visual channel. This is not surprising as the dataset is collected from a visual experiment, so most activity is visual. ChannelGPT2 closely matches both the amplitude and the timing of the evoked response peaks across the whole 1-second epoch. Variability across trials is also well matched.

To quantify the similarity between real and model generated evoked activity, we computed the correlation of the mean (across individual epochs) time-courses of the evoked response for each channel separately. Note that we averaged over the different MEG sensors (the magnetometers and gradiometers) found at the same location. The result of this is plotted in Figure 6, allowing insights into the spatial pattern of similarity.

As expected, ChannelGPT2 generates data with evoked responses that have much higher correlation with evoked responses from real data, and slightly higher correlation in visual ar-

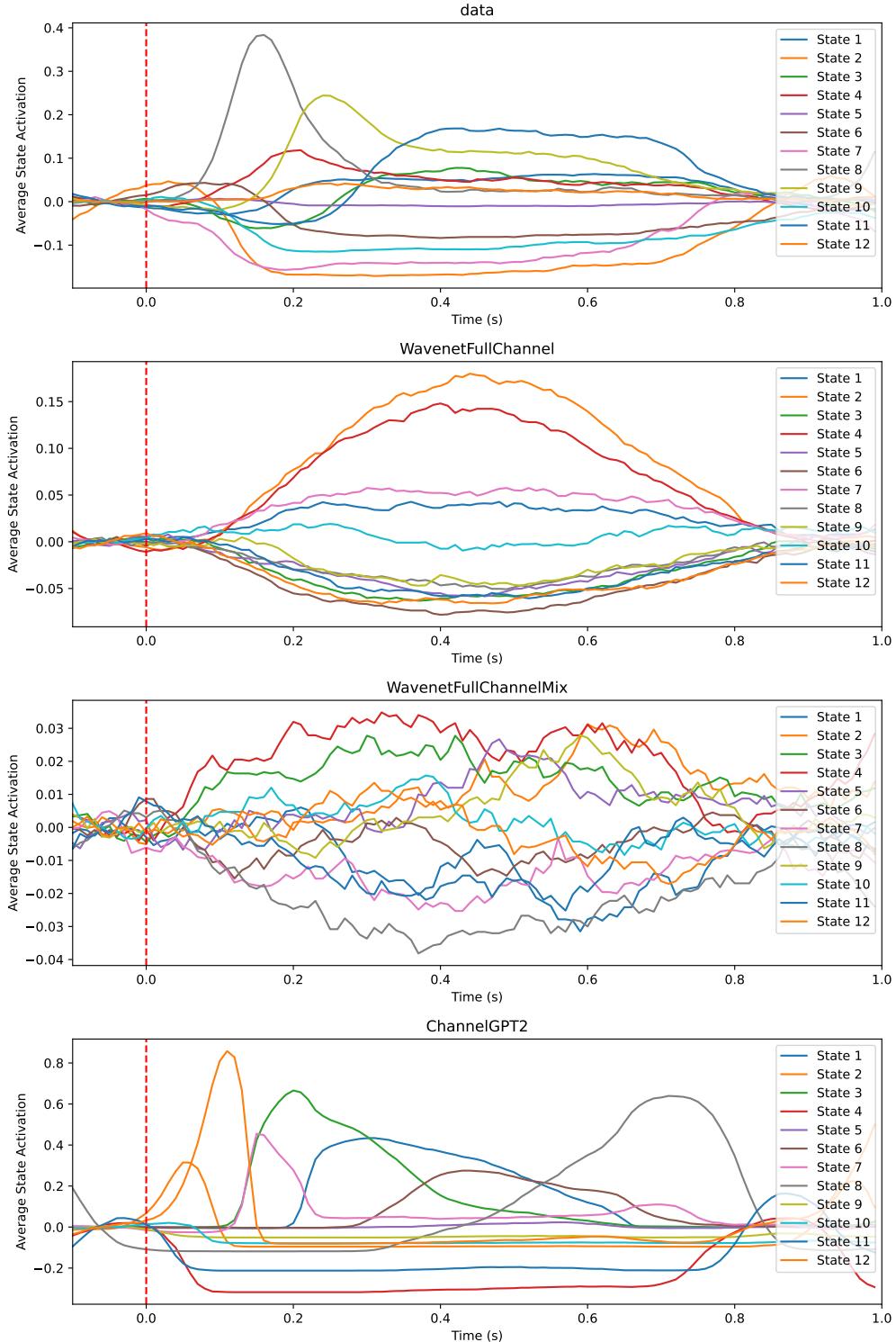


Figure 8: Evoked response state timecourses of HMMs trained on the real MEG data and generated data from each of our task-conditioned models. Note that states are not matched between models. Image presentation starts at 0 seconds and ends at 0.5 seconds.

eas compared to other channels, matching the known topography of visual evoked responses. In other models the correlation is low, and spatially better in frontal areas, likely because the evoked responses here are noisier providing an easier fit.

Figure 7 shows the correlation between the variance (over individual epochs) time-courses of the mean evoked response obtained from the actual data and the evoked responses obtained from data generated by each model. This captures a measure of the ability of the models to represent the trial-to-trial variability found in the real data. Again, ChannelGPT2 generates data that has the highest correlations with the real data, with higher values in channels in the back of the head, appropriately capturing the topography of response variability. Other models have similar spatial distribution, and notably WavenetFullChannel also produces evoked responses with variance partially matching the real data.

Finally, a different way to assess task-related activity is to examine the evoked state time-courses from the HMMs fitted on the real and model generated timeseries. Rather than looking at individual channels, this provides an overall view of which HMM state gets activated when, during individual trials. This is computed by simply epoching the state timecourse, and averaging over all trials. We plot these for the real data and each generated timeseries in Figure 8. As expected, the HMM trained on models other than ChannelGPT2 shows poor evoked state time-courses. ChannelGPT2 generated data produces states with similar evoked dynamics and variability as the real data.

2.5 Group-level ChannelGPT2 adapts generated data to individual subjects

Up to this point, all trainings and analyses were done on MEG data from a single subject. We next looked at whether adding data from multiple subjects improves modelling and generation performance. This is in line with the overall goal of training such foundational forecasting models on multiple large datasets. Here we took a first step in exploring this by scaling ChannelGPT2 to the 15 subjects in the Cichy et al. (2016) data, which we refer to as ChannelGPT2-group. For adapting to multiple subjects and to capture variability over subjects, we used subject embeddings as described in Section 3. The main reason for only evaluating ChannelGPT2 on group data is the comparatively much poorer performance of Wavenet-based models in evoked timeseries generation.

We were interested in whether the model generated evoked responses improved their similarity with the evoked responses from the real data, when using data from more subjects. To compare with the single-subject training we generated data using the subject embedding of that subject. The comparison of the evoked response of single-subject and group models for one 1 visual channel is shown in Figure 9. We found that generally ChannelGPT2-group produces evoked responses that are more smoothed than the single-subject model. This is possibly because the model learns to generate data that is closer to the average statistics over subjects, and while it can adapt its generation based on the subject label — this ability is not perfect.

To test our hypothesis regarding ChannelGPT2-group generating more of an average across subjects, we generated data for all subjects (using appropriate subject embeddings) and compared the grand average evoked responses with those extracted from the MEG data of all subjects. Two channels are plotted in Figure 10. The evoked response averaged over all subjects is much noisier because of the high between-subject variability. However, we can see that indeed ChannelGPT2-group can generate this well, perhaps slightly smoother

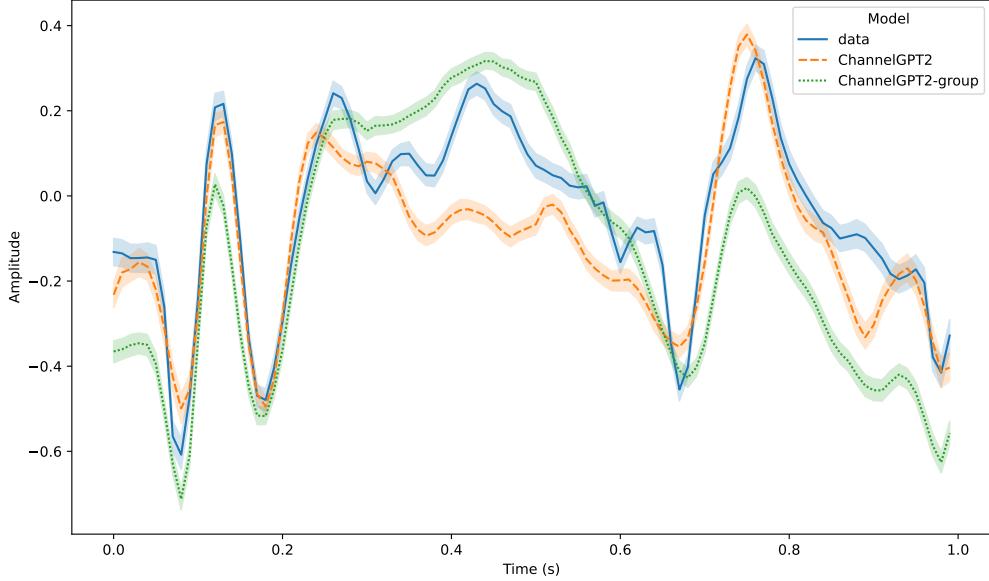


Figure 9: Comparison of evoked responses in a visual channel (MEG2332) across single subject and group models. The stimulus onset is at 0 s and the stimulus offset is at 500 ms. Shading indicates 95% confidence interval across trials.

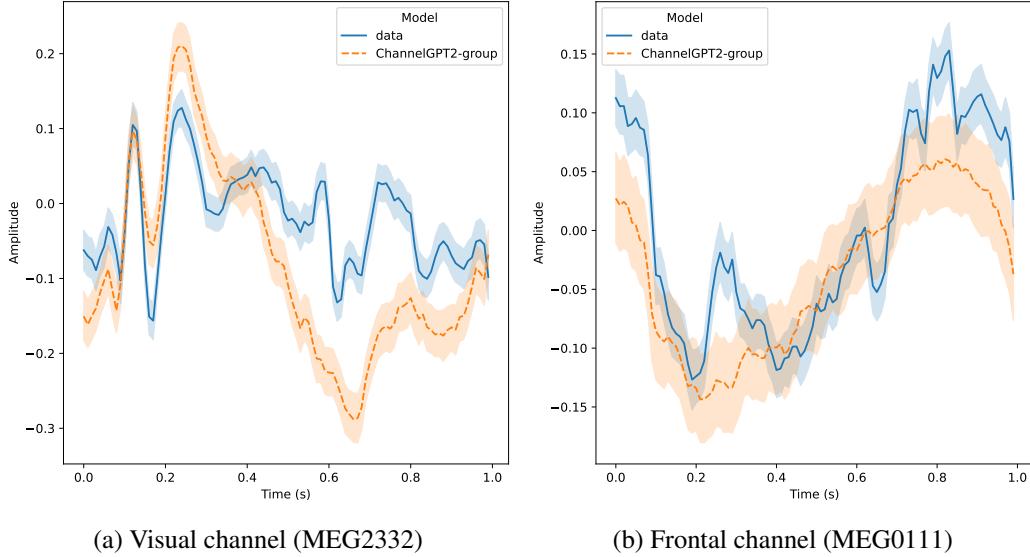


Figure 10: Comparison of evoked responses averaged across all subjects in the data (blue line) and the generated data from ChannelGPT2-group (orange line). The horizontal axis encompasses 1 second, stimulus onset is at 0 seconds and stimulus offset is at 0.5 seconds. Shading indicates 95% confidence interval across trials.

than the real data. Comparing these plots with Figure 9, it is also clear that it adapts its generation well to a specific subject compared to the group average.

A further way to test alignment between group-level evoked responses is to fit an HMM on the data of all subjects, and then infer state timecourses with this model on the generated

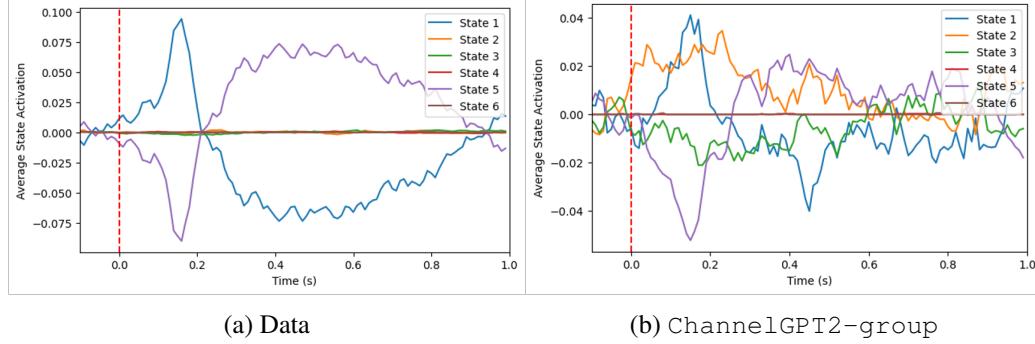


Figure 11: Comparison of evoked state timecourses inferred from the data of all subjects and from the generated data of ChannelGPT2-group for all subjects. State indices are matched between the two plots, as the same fitted HMM model was used.

data of all subjects from ChannelGPT2-group. By taking this approach we can directly match the evoked state timecourses between the real and generated timeseries. We trained an amplitude-envelope HMM (AE-HMM) with 6 states (Quinn et al., 2019) and show results in Figure 11. Two states that show strong activation during real task data show similar temporal signatures and amplitude changes in the generated data, albeit slightly noisier. In the generated data there are two additional states which seem to get activated during the trial. This indicates that while ChannelGPT2-group can capture some of the state-level dynamics, there is room for improvement.

Finally, we examine the variability in state time courses over individual trials. For this we trained an 8-state HMM on the real data of a single subject, and inferred the state timecourses on both the single-subject ChannelGPT2 and ChannelGPT2-group generated data (with the appropriate subject embedding), obtaining matched states. We hypothesised that even if the average evoked responses are similar to the real data, GPT2 may not be able to generate trials with variability in the temporal activation of states. Figure 12 shows that this is indeed true for the single-subject ChannelGPT2 generated data. ChannelGPT2-group responses seem to include much higher temporal variability in state activations, though still falling short of the real data. This indicates that the model can capture some trial-to-trial variability through its exposure to multiple subjects, but has difficulty fully matching the complexity of real neural data. More data may be needed to improve this aspect of generation.

2.6 Group-level ChannelGPT2 generates classifiable evoked responses

We have shown that deep learning models, and particularly the channel-independent Transformer-based model (ChannelGPT2), can generate data with spatial, temporal, and spectral signatures similar to real data. We were next interested whether such a foundational model can aid in a downstream task. Specifically, we look at the ability of ChannelGPT2 to aid in the decoding of experimental task conditions in Cichy et al. (2016).

We first investigated whether the task responses generated by the ChannelGPT2 model can be classified with performance comparable to trials of real data. This also further tests how well the model captures spatiotemporal task-related activity and information. The benefit of this approach is that if similar performance is obtained, then ChannelGPT2 could simulate

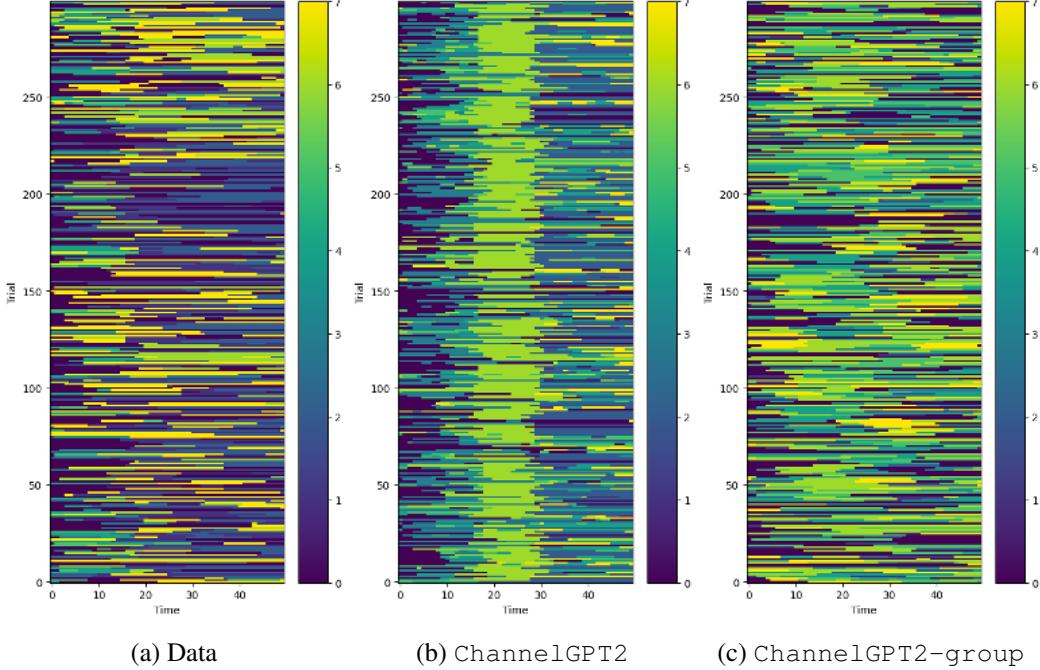


Figure 12: Comparison of trial-level variability in the evoked state timecourses of an HMM trained on real data and applied to the generated data of ChannelGPT2 and ChannelGPT2-group. Different colours represent different states (matched across models). Individual trials however are not matched and we cannot compare the plots at the trial-level, only as an aggregate visualisation of variability across trials.

an arbitrarily large number of trials to potentially improve decoding of real data through pretraining on this simulated data. This is a form of transfer learning where the decoding model, not the forecasting model, is transferred.

We generated 20 trials for all 118 conditions for 1 subject with both ChannelGPT2 and ChannelGPT2-group. We trained separate linear neural network models on the real data (20 trials/condition) and the generated datasets, with an appropriate 4:1 train and validation set ratio. This achieved 17.6%, 1.9%, and 7.2% validation accuracy for the real data, ChannelGPT2, and ChannelGPT2-group, respectively. Thus the group model generates more classifiable subject-specific task-responses, but still does not reach the classifiability of real data. This and previous analyses indicate the group model successfully leverages larger datasets to produce more accurate task-related activity.

We also tried obtaining a decoding model directly from the ChannelGPT2-group forecasting model using Bayes’ theorem. We found limited 5% accuracy over 1 subject’s validation set (versus 40-50% with a discriminative decoder). This generative decoding approach may require larger datasets or more sophisticated architectures.

2.7 Transfer learning

A key advantage of generated data is the ability to generate huge amounts of surrogate data. We generated additional datasets with 40 and 60 trials/condition using ChannelGPT2-group. Training a decoder on these achieved 21.7% and 44.2% ac-

curacy, respectively, exhibiting linear scaling of classification performance with simulated data amount. Critically, we assessed whether this simulated data can pretrain classifiers for transfer learning. We first pre-trained the neural network decoder on the 20-, 40-, and 60-trial generated datasets, then finetuned it (trained it further) on the real MEG dataset (20 trials/condition). As the simulated data used for pre-training increased, accuracy of the finetuned model improved rapidly. Zeroshot (no finetuning) performance on real data was above chance with 2%, 3%, and 4% accuracy, for increasing pretraining data quantities. Final accuracies after finetuning were 19.5%, 21.5%, and 23%, respectively. Thus, each additional 20 simulated trials/condition improved final decoding by ~2%. These results are summarised in Table 3.

2.8 Ablation experiments

Ablation studies are a common approach in machine learning to understand model behaviour by selectively removing or altering components of the model (Meyers et al., 2019). We performed ablation experiments with ChannelGPT2 to investigate how well it can generate task-related brain activity under varied conditions without further training.

First, we evaluated the model’s ability to adapt to different trial durations. The results reported thus far are for a ChannelGPT2 trained on trials lasting 0.5 seconds. We generated data using the same fitted ChannelGPT2 model but with trial durations of 0.2 s and 0.8 s. As shown in Figure 13, ChannelGPT2 accurately adapted to the shorter and longer trials. The evoked responses matched the expected time-courses, with appropriate truncation or lack of second peaks due to stimulus offset. This demonstrates the model’s ability to generalise to varied trial durations despite being trained on a fixed duration.

Next, we performed two experiments to determine whether ChannelGPT2 relies solely on timing information or also utilises the semantic content of the condition labels. First, we trained a model (ChannelGPT2-randomlabel) where the condition labels were shuffled randomly during training, breaking the semantic alignment between labels and evoked responses. Second, we trained a model (ChannelGPT2-1label) using a single

Trained on (no. trials)	Tested on MEG (20)	Tested on GPT2 (same no. trial data)
MEG (20)	17.6	-
GPT2 (20)	2	7.2
GPT2 (40)	3	21.7
GPT2 (60)	4	44.2
GPT2 (20) + MEG (20)	19.5	-
GPT2 (40) + MEG (20)	21.5	-
GPT2 (60) + MEG (20)	23	-

Table 3: Summary of transfer learning results. The first column shows the data used for training the decoder, with the number of trials per condition shown inside the parenthesis. GPT2 refers to the ChannelGPT2-group generated data, while GPT(.) + MEG (20) is the fine-tuned decoder on the MEG data. The other two columns represent the validation data on which the decoder performance is shown. Accuracy values are provided in percentages. Chance level is 100/118.

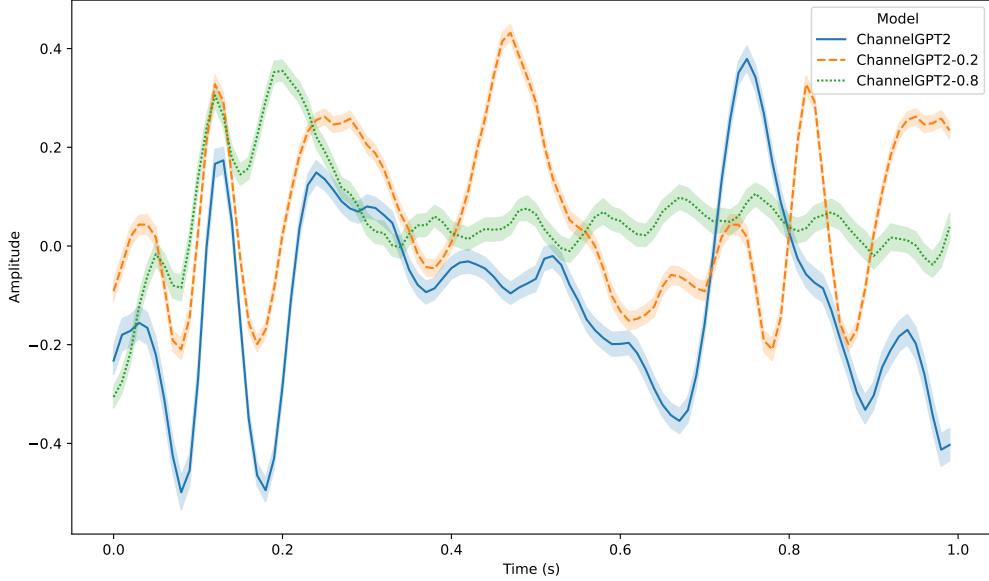


Figure 13: Evoked responses generated by ChannelGPT2 for trials of 0.2 s (orange), 0.5 s (blue), and 0.8 s (green). In all cases, the model was trained only on data containing trials of 0.5 s, but can then adapt appropriately to the different durations. The plotted channel is in the visual area (MEG2332).

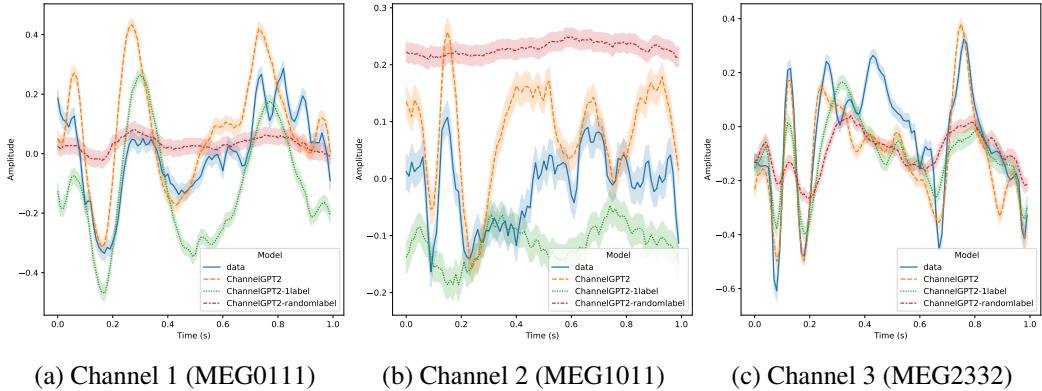


Figure 14: Evoked responses for ChannelGPT2 models trained with shuffled or single condition labels, indicating reliance on semantic content. Three representative channels are presented. MEG0111 is anterior-left, MEG1011 is anterior-central, and MEG2332 is posterior-central. See main text for an explanation of model types. Stimulus onset is at 0 seconds, with stimulus offset at 0.5 seconds.

condition label for all trials. This tests whether the model cheats by learning an average evoked response instead of adapting to each condition.

As evident in Figure 14, both models failed to generate distinct evoked responses for different semantic conditions. This demonstrates that ChannelGPT2 leverages both timing and semantic information in the conditioning labels, rather than simply learning a stereotyped temporal template. Quantitatively, evoked response correlation with real data dropped to 44% and 56% for ChannelGPT2-randomlabel and ChannelGPT2-1label, re-

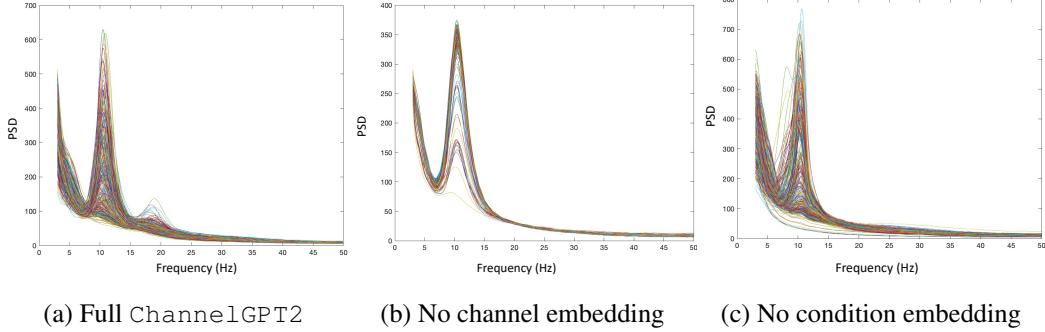


Figure 15: Generated power spectra for full ChannelGPT2 model (left) versus ablations. Both channel (middle) and condition embeddings (right) are critical for accurate spectral content.

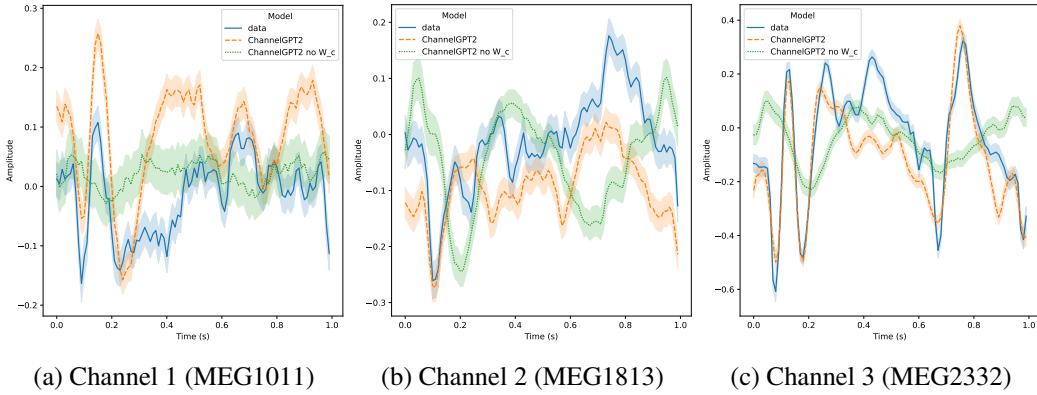


Figure 16: Comparison of generated evoked responses from ChannelGPT2 and the model with ablated channel embeddings (ChannelGPT2 no W_c) across 3 representative channels. Without channel embeddings the model fails to adapt evoked responses to different channels. The stimulus onset is at 0 seconds and the offset is at 0.5 seconds.

spectively, compared to 74% for the full ChannelGPT2. Both the qualitative analysis and the correlation numbers indicate that ChannelGPT2-1label was somewhat closer to matching ChannelGPT2.

We also investigated the contributions of the channel and condition embeddings, by training two separate ablated models. As shown in Figure 15, removing the channel embeddings resulted in very similar PSD across channels in the generated data, indicating the model relies heavily on these embeddings to adapt generation per channel. The evoked responses in Figure 16 confirm that without channel embeddings, variability between channels is reduced. Removing the condition embeddings resulted in noisier power spectra of the generated data and no 20 Hz peak.

Finally, we found that the channel embeddings encode spatial relationships, as sensors that are near to each other in the real sensor montage tend to have more similar embeddings. This is shown through a t-SNE and PCA projection of the embedding space in Figure 17. Correlation between pairwise Euclidean distances of channels in physical space and embedding space was 0.45 (Figure 17c).

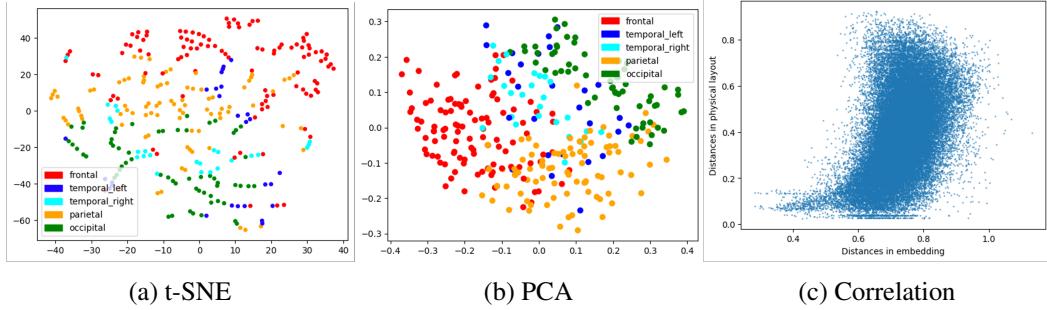


Figure 17: (a) (b) 2D projection of the channel embeddings from ChannelGPT2-group with t-SNE and PCA. Channels are coloured by their location on the scalp grouped into 5 major brain areas. (c) Plotting pairwise Euclidean distances of channels in real, physical space versus embedding space. Sensors that are near to each other in the real sensor montage tend to have more similar embeddings. Each point represents a different pair of channels. Correlation is 0.45.

3 Methods

3.1 Multi-channel Wavenet

Here we describe how we adapted the Wavenet architecture (van den Oord et al., 2016) for electrophysiological data. Wavenet models the conditional probability of each time sample given all preceding samples autoregressively:

$$p(\mathbf{X}) = \prod_{t=1}^T p(\mathbf{x}_t | \mathbf{x}_1, \dots, \mathbf{x}_{t-1}) \quad (1)$$

where \mathbf{x}_t is the sample at time t and T is the total sequence length. The network predicts a categorical distribution over tokenised samples using a softmax output layer. Throughout this paper we use tokenisation and quantisation interchangeably. Both have the aim of discretising a continuous quantity into a finite set of distinct bins/levels/tokens.

In the original paper, the audio waveform is tokenised using a quantisation to 8 bits following a μ -law companding transform (Lewis and MTSA, 1997):

$$f(\mathbf{x}_t) = \text{sign}(\mathbf{x}_t) \frac{\ln(1 + \mu|\mathbf{x}_t|)}{\ln(1 + \mu)} \quad (2)$$

where μ controls the number of quantisation levels, set to 255 as in the original Wavenet. $f(\cdot)$ is applied to each value of \mathbf{x}_t independently. This nonlinear transformation improves reconstruction versus uniform quantisation of the raw input, as it skews the distribution such that more levels are allocated to smaller magnitudes. For MEG data, we observe similar benefits when applying this transform prior to quantisation. Note that the input must be scaled to $(-1, 1)$ first, and clipping outliers above some threshold helps ensure a more uniform mapping.

Critically, tokenisation, in this case through quantisation, enables modelling of probability distributions over data and sampling, instead of just point estimates from MSE-based training.

Cross-entropy loss also avoids the mean-prediction bias induced by MSE (Banville et al., 2021).

When adapting Wavenet to M/EEG, a key challenge is the multi-channel nature of the data. We devise two versions: `WavenetFullChannel` as univariate, and `WavenetFullChannelMix` as multivariate. In both, each channel is transformed and tokenised independently to form the input to the models.

In `WavenetFullChannel`, we first apply an embedding layer to the tokenised data, learned separately per channel. The embedding layer represents each discrete bin as a high-dimensional continuous vector, enabling powerful representations in the convolutional layers whose input channels match the embedding size. To be clear in this univariate approach the same model is applied to each channel. However, a different embedding layer is learned for each channel, meaning that for example the quantised value of 0.42 in channel x will have a different vector representation than in channel y. This helps the model differentiate between channels.

The embedding operation is given below:

$$\forall c \in 1, 2, \dots, C : \mathbf{X}_e^{(c)} = \mathbf{W}^{(c)} \mathbf{X}^{(c)} \quad (3)$$

$$\mathbf{H}_0 = \text{Concatenate}(\mathbf{X}_e^{(1)}, \mathbf{X}_e^{(2)}, \dots, \mathbf{X}_e^{(C)}) \quad (4)$$

Here, $\mathbf{X}^{(c)} \in \mathbb{R}^{Q \times T}$ is the tokenised one-hot input and $\mathbf{W}^{(c)} \in \mathbb{R}^{E \times Q}$ is the embedding layer of channel c mapping tokens Q to embeddings of size E . Concatenate concatenates along the channel dimension.

$\mathbf{H}_0 \in \mathbb{R}^{C \times E \times T}$ is the resulting input to Wavenet with C as the batch dimension. Thus, the same model is applied independently to each channel in parallel. At output, a distribution is predicted simultaneously for each channel at $T + 1$. The model is optimised to accurately predict all channels.

`WavenetFullChannelMix` includes an extra linear layer after summing the skip representations to mix information across the channel dimension:

$$\mathbf{S} = \sum_{l=1}^L \mathbf{S}^{(l)} \quad (5)$$

$$\mathbf{S} = \mathbf{S}.\text{permute}(1, 2, 0) \quad (6)$$

$$\mathbf{S}_{out} = \mathbf{S}\mathbf{W}_m \quad (7)$$

where $\mathbf{W}_m \in \mathbb{R}^{C \times C}$ is the mixing weight matrix, and $\mathbf{S}^{(l)}$ is the output of the skip connection at layer l . The permutation is needed to apply the projection to the appropriate channel dimension. After this \mathbf{S}_{out} is permuted back to the original dimension order and the rest proceeds identically to `WavenetFullChannel`.

In the original Wavenet, audio generation can be conditioned on additional inputs through embedding-based global conditioning or time-aligned local conditioning. For some experiments, we augment the model with local features of task stimuli or subject labels, first embedded into continuous vectors:

$$\mathbf{H}_y = \mathbf{Y}\mathbf{W}_y \quad (8)$$

$$\mathbf{H}_o = \mathbf{O}\mathbf{W}_o \quad (9)$$

$$\mathbf{H}_c = \text{Concatenate}(\mathbf{H}_y, \mathbf{H}_o) \quad (10)$$

where $\mathbf{Y} \in \mathbb{R}^{T \times N}$ contains the condition index $n \in (1, \dots, N)$ at each time point, and $\mathbf{O} \in \mathbb{R}^{T \times S}$ contains the subject index $s \in (1, \dots, S)$ at each time point $t \in (1, \dots, T)$. $\mathbf{W}_y \in \mathbb{R}^{N \times E_n}$ and $\mathbf{W}_o \in \mathbb{R}^{S \times E_s}$ are embedding matrices mapping the labels to learned continuous vectors of size E_n and E_s , respectively. The subject index is the same across time points of the recording from the same subject. The condition index is set to the (visual) stimuli presented (e.g., one of the 118 images in Cichy et al. (2016)), for exactly those time points when the stimulus is on. At any other time, the task condition embedding \mathbf{H}_y is set to 0.

\mathbf{H}_c is the conditioning vector fed into Wavenet at each layer:

$$\mathbf{Z}^{(l)} = \tanh \left(\mathbf{W}_f^{(l)} * \mathbf{H}^{(l)} + \mathbf{W}_c^{(l)} * \mathbf{H}_c \right) \odot \sigma \left(\mathbf{W}_g^{(l)} * \mathbf{H}^{(l)} + \mathbf{W}_c^{(l)} * \mathbf{H}_c \right) \quad (11)$$

where $\mathbf{W}_c^{(l)}$ (1x1 convolution) projects \mathbf{H}_c before adding it to the input representation ($\mathbf{H}^{(l)}$). $\mathbf{W}_f^{(l)}$ is the filter convolution weight, $\mathbf{W}_g^{(l)}$ is the gate convolution weight, and $\mathbf{Z}^{(l)}$ is the output representation at layer l . \odot is element-wise multiplication. This formulation conditions the prediction on both past brain activity and stimuli:

$$p(\mathbf{X}|\mathbf{Y}, \mathbf{O}) = \prod_{t=1}^T p(\mathbf{x}_t | \mathbf{x}_1, \dots, \mathbf{x}_{t-1}, \mathbf{y}_1, \dots, \mathbf{y}_{t-1}, \mathbf{o}_1, \dots, \mathbf{o}_{t-1}) \quad (12)$$

In single-subject models we only use the task labels \mathbf{Y} .

The full Wavenet architecture can either be interpreted as forecasting with extra conditioning or as a generative encoder augmented with past brain activity. In addition, the probabilistic formulation allows converting the model into a decoder using Bayes' rule, enabling both forecasting and decoding within the same framework:

$$p(Y|X) = \frac{p(X=x|Y)p(Y)}{p(X=x)} \quad (13)$$

where X is the random variable representing the data, Y is the random variable representing the task labels, and x is a particular sample of X . $p(Y)$ is the task label prior distribution which in the 118-image dataset is uniform. $p(X=x|Y)$ is the likelihood of the data given the label which we get from the above formulation of Wavenet. The only tricky part is $p(X=x)$ as this requires marginalisation over Y . In the case of the 118-image dataset this means that we have to run the trained model with all of the possible task labels to obtain $p(X=x)$:

$$p(X = x) = \sum_{i=1}^N p(X = x|Y = i)p(Y = i) \quad (14)$$

Thus, in a single self-supervised deep learning model we have flexibly encapsulated forecasting, encoding, and decoding, all three of the main modelling methods of M/EEG data. This unification of modelling approaches was inspired by a GitHub repository applying similar ideas to images². The inverted decoder formulation also allows for iterative estimation of $p(Y|X)$ at each timestep. The author of the GitHub repository has applied this method to estimating the probability of image labels (digits 0 to 9) from pixel images, as more and more of the image was fed into the model.

3.2 Multi-channel GPT2

We set out to design a Transformer model suited for M/EEG data, while keeping the key elements that made it successful in language modelling. Specifically, we use GPT-2, a popular autoregressive Transformer variant. When adapting GPT-2 to continuous multivariate time series, the main challenges are at the input and output layers interfacing the model with the data. A particularly detailed visual description of GPT2 is given in Alammar (2019).

Altogether a GPT2 layer is the combination of the self-attention and feedforward layers:

$$\mathbf{H}^{(0)} = \mathbf{X}\mathbf{W}_e + \mathbf{W}_p \quad (15)$$

$$\mathbf{Z}^{(l)} = \text{LN}(\mathbf{H}^{(l)} + \text{MultiHeadAttention}(\mathbf{H}^{(l)})) \quad (16)$$

$$\mathbf{H}^{(l+1)} = \text{Dropout}(\text{LN}(\mathbf{Z}^{(l)} + \text{FFN}(\mathbf{Z}^{(l)}))) \quad (17)$$

$$\hat{\mathbf{Y}} = \text{softmax}(\mathbf{H}^{(L)}\mathbf{W}_e^T) \quad (18)$$

where $\mathbf{W}_e \in \mathbb{R}^{Q \times E}$ embeds the discrete tokens $\mathbf{X} \in \mathbb{R}^{T \times Q}$ into E dimensions. LN is Layer Normalisation, a regularisation technique which normalises all activations within a layer to zero mean and unit variance. $\mathbf{W}_p \in \mathbb{R}^{T \times E}$ contains positional encodings, providing the model with sequential order information. This is needed as GPT2 lacks recurrent or convolutional elements. Each vector in \mathbf{W}_p indexed by $t \in (1, \dots, T)$ contains a distinct E -dimensional representation of position t . The output $\mathbf{H}^{(L)}$ is projected back to the vocabulary via the transpose embedding matrix (weight tying). Alternatively, a separate output projection can be learned. The softmax output gives a token probability distribution.

GPT-2 is trained via supervised learning to predict the next token given previous context, minimising cross-entropy loss between model outputs $\hat{\mathbf{Y}}$ and ground truth targets \mathbf{Y} . To enable autoregressive training, \mathbf{Y} is set to \mathbf{X} shifted one timestep ahead. Crucially, to prevent information leakage from future timesteps $t + 1, \dots, T$, causal masking is applied in each self-attention layer, setting outputs that would reveal future information at position t to zero.

To apply GPT2 to our continuous multichannel time series data, we take a similar approach as with Wavenet by tokenising each channel independently using the same method as before. This serves as our equivalent of the discrete set of tokens in language modelling. The same

²<https://github.com/cheind/autoregressive>

GPT2 model is applied to each channel in parallel by setting the channel dimension as the batch dimension. We call this ChannelGPT2.

The input to the model includes the position embedding as well as subject and task-stimulus embeddings. We also add a label/embedding telling GPT2 which channel the current time series is coming from:

$$\mathbf{H}^{(0)} = \mathbf{X}\mathbf{W}_e + \mathbf{W}_p + \mathbf{Y}\mathbf{W}_y + \mathbf{O}\mathbf{W}_o + \mathbf{W}_c \quad (19)$$

where $+$ denotes element-wise addition, $\mathbf{X} \in \mathbb{R}^{C \times T \times Q}$ is the tokenised input, $\mathbf{W}_c \in \mathbb{R}^{C \times T \times E}$ are the learned channel embeddings of size E , which are distinct for each channel $c \in 1, \dots, C$ but constant across time t . \mathbf{Y} and \mathbf{O} are the task and subject index matrices, mapped to their respective embeddings. As with the positional encoding \mathbf{W}_p , we simply add all embeddings (task, subject, channel) into a single representation. Note that instead of having channel-specific embeddings of the tokenised input \mathbf{X} we learn the same mapping $\mathbf{W}_e \in \mathbb{R}^{Q \times E}$ across channels. Channel information is provided to the model through the channel embeddings.

A serious limitation of this channel-independent GPT2 model is that when predicting a single channel, it does not receive information from other channels. This is analogous to a univariate autoregressive model and ignores crucial cross-channel dependencies in the data. To be clear we often use the term univariate AR modelling in the sense that a separate AR model is trained on each channel. In the case of channel-independent Wavenet and GPT2 models, we train one and the same model on all channels.

3.3 Model interpretation

To evaluate whether Wavenet and GPT2 models accurately capture brain dynamics beyond just predictive performance, we develop several analysis techniques to interrogate what these models learn.

Data generation Different models have distinct generation procedures. Linear AR models take Gaussian noise as input and generate one timestep at a time. Gaussian noise is added to the output, which is appended to the input sequence. This recursive process is described by:

$$\mathbf{x}_t = \epsilon_t + f(\mathbf{X}_{t-K:t-1}) \quad (20)$$

This intuitively treats the model f as a black-box infinite impulse response (IIR) filter, where $\epsilon_t \sim \mathcal{N}(0, 1)$, and K is the receptive field size.

For tokenised models (Wavenet and GPT2), we generate data by sampling from the predicted probability distribution and recursively feeding the sample back as input. We use top-p sampling, which samples from the ordered outputs whose cumulative probability mass exceeds $p\%$ (Holtzman et al., 2020). top-p sampling often works better in practice than using the full distribution by avoiding generation of low-probability tokens, and thus reducing noise.

Comparison metrics We compare generated timeseries to real data using power spectral density (PSD), and Hidden Markov Model (HMM) statistics. We fit a separate 12-state time-domain embedding HMM (TDE-HMM) to each multivariate generated timeseries (Vidaurre et al., 2018b). We used the `osl-dynamics` package (Gohil et al., 2023), and set the number of embeddings to 15, the PCA projection dimensionality of the channels to 80 and the sequence length to 2000. We trained the HMMs for 20 epochs with an initial learning rate of 0.02. Once fit we compare the state timecourses of generated timeseries and real data. We also compute four summary statistics (fractional occupancy, mean lifetime, mean interval, switching rate), and compare the distributions of these statistics across the 12 states. Finally we also compare the power spectra of the timeseries corresponding to each state.

Evoked activity For task-conditioned models, we assess reconstruction of task-dependent dynamics by feeding in task labels during generation and examining evoked responses. We compare visually average evoked responses from generated and real data, and also quantify the correlation between generated and real evoked responses. Furthermore, we epoch the state timecourses of the HMMs fit to generated and real data and compare evoked state timecourses. The variability in the evoked state timecourses is also compared.

Classification To evaluate how well models capture task activity, we apply standard decoding models (e.g., linear classification) to generated trials and compare performance to real data. We also evaluate the generalisability of decoders trained on generated data to real data. Strong similarity in these metrics would indicate accurate modelling of task responses. For decoding both generated and real data we use a simple four-layer linear neural network introduced by (Csaky et al., 2023a). The first layer of the model performs a dimensionality reduction across channels.

Ablation By removing certain model components and evaluating performance, ablation studies assess the contribution of different architectural factors. We perform ablations on conditioning and channel embeddings.

3.4 Model and training details

As our dataset of choice, we used the continuous 15-subject, 118-image data from Cichy et al. (2016). Here each subject views 118 distinct images, with 30 trials/image. For each subject, the data was bandpass filtered between 1 and 50 Hz, and a notch filter was applied to remove line noise. Subsequently, independent component analysis (ICA) artifact rejection was performed with a dimensionality of 64. Components were visually inspected for each subject, and those that exhibited clear artefactual features (e.g. eye or cardiac signals) were removed. The data was then downsampled to 100 Hz. The continuous data was split into non-overlapping validation, test, and training sets. The validation and test sets included 4 trials of each of the 118 conditions, while the training set contained the remaining 22 trials. This non-overlapping uniform splitting of the continuous data was possible due to the experimental setup during data recording.

For each model the data from each channel was tokenised independently to 256 bins using a quantisation via the mu-law algorithm discussed in Section 3.1. To achieve uniform quantisation, we first standardised each continuous-data channel, clipped values higher than

4 or lower than -4, applied per-channel maximum absolute scaling to map the data to the range (-1, 1), and finally applied the mu-law transform and 8-bit quantisation.

Our aim was to evaluate several models and methods on this dataset. Due to computational constraints and limited iteration speed over experiments and methods, all experiments were performed on a single representative subject, except in Section 2.5 where we explore our GPT2 model on all 15 subjects, and the classification and transfer learning analyses in Section 2.6 and Section 2.7.

We trained univariate AR(255) models on each channel. The AR order was chosen to match the receptive field of the other models. We implemented and trained the AR model using a single linear convolutional neural network layer. Note that we did also assess multivariate AR models (results not shown), but this did not improve performance compared to the univariate AR. We trained `WavenetFullChannel` with a matched receptive field of 255, two stacks of dilation blocks (7 layers per block, doubling dilation factors), 256 hidden channels, 1024 skip channels, no dropout, and a 20-dimensional task embedding. `WavenetFullChannelMix` had the same architecture but 128 hidden channels and 512 skip channels. We used early stopping on the validation set. This means that we ran training until overfitting was observed, and then analysed the model version with the lowest validation loss. All our analyses were performed on the distinct test set.

Our Channel-independent GPT2 (`ChannelGPT2`) had a variable receptive field between 128 and 256. This means that during training the model encountered examples that had a sequence length between 128 and 256, rather than all examples having the same length. GPT2 is normally trained to output all timesteps in a sequence of length T , given previous timesteps. However, this means that for the second timestep, the receptive field is only 1. Ideally, we wanted to match the training setup of our Wavenet models, where the receptive field is always 256. However, this would significantly slow down training as the whole forward and backward pass must be recomputed at each timestep. We opted for a trade-off, where we set the minimum receptive field to 128, ensuring efficient training and that the model is not trained to predict shorter sequence lengths.

The embedding size of all inputs (token vocabulary, position, task, channel) was set to 96, and we used 12 GPT2 layers, with 12 attention heads. We used Huggingface’s implementation³, so the rest of the parameters were the same as in their configuration. Dropout was set to 0 and we used early stopping on the validation set.

On average the mu-law quantisation achieved low reconstruction error. We tested the reconstructed data by performing evoked analysis, and classification of the task responses, and achieved comparable performance to the raw data (results not shown). Thus, both types of tokenisation add negligible quality loss to the data.

For group trainings we used the same hyperparameters as for the single-subject trainings, except for increasing the embedding size to 240. `ChannelGPT2-group` proved difficult to overfit, meaning that using more data acted as a regulariser, and we stopped training when validation losses did not improve for 5 consecutive epochs.

³<https://github.com/huggingface>

4 Discussion

We have presented our initial efforts at developing a general forecasting model for M/EEG data. After carefully evaluating the trade-offs between various modelling approaches, we settled on two main architectures: one based on Wavenet (van den Oord et al., 2016), and one based on GPT-2 (Radford et al., 2019). These models have proven successful in the audio and natural language domains, which share similarities with the time series nature of brain signals. We systematically compared different variants of our proposed models on a visual-task MEG dataset.

We found that the forecasting performance was comparable between Wavenet and AR models according to next-timestep prediction metrics. This suggests such metrics may be limited in their ability to effectively evaluate model dynamics beyond one-step prediction. Perhaps looking at these metrics when recursively generating multiple timesteps in the future might be more informative. Generated data analysis provided more discerning model comparisons. While the channel-independent AR and Wavenet models accurately reproduced the overall power spectral density, only the Transformer-based models captured more abstract multivariate statistics like HMM state dynamics.

Critically, the ChannelGPT2 model-generated data closely matched real MEG recordings across both temporal and spectral domains. Analysis of the discovered latent brain states showed ChannelGPT2 reproduced variable oscillatory states similar to those inferred from human recordings (Vidaurre et al., 2018a). Each state captured distinct spectral content, while the linear and Wavenet-based models failed to achieve this degree of heterogeneity in their dynamics. It is possible that this does not indicate a failing of the Wavenet architecture, but rather that different conditioning methods may be needed. One such approach that we have not tested is using the same type of channel embeddings as for ChannelGPT2.

Multiple analyses consistently demonstrated ChannelGPT2’s strengths in realistic conditional timeseries generation. ChannelGPT2-generated evoked responses had high correlation to real MEG data. However, modelling single-trial variability and between-subject differences remain challenging areas needing further work. Scaling to multiple subjects showed promise. The model was able to adapt its generated data based on the input subject label and generate task trials with variability more similar to real recordings than a single-subject model.

Ablation studies quantified the importance of channel embeddings and task conditioning for accurate MEG modelling. Removing channel embeddings resulted in near identical generation across sensors, failing to capture spatial heterogeneity. Analysis of ChannelGPT2’s channel embeddings revealed spatial relationships between sensors were learned, with proximal channels having more similar embeddings. With incorrect or with no task labels, ChannelGPT2 produced noisy evoked responses, indicating the model leverages both timing and label semantics. Furthermore, the model trained on 0.5s trials only, was able to produce reasonable responses to longer or shorter trials, showcasing generalisation. These results demonstrates the value of multi-faceted conditioning for realistic brain data modelling.

A key investigation involved analysing the classification of generated data according to the condition labels. The trials generated by the group-level model were classified with much higher accuracy (closer to real data) than those of the single-subject model. We

also demonstrated that generated data can improve decoding of real trials via transfer learning (Torrey and Shavlik, 2010), with benefits scaling with generated data quantity. The classifiability of generated trials and transfer learning results highlight the utility of forecasting models like ChannelGPT2 for decoding real MEG data. Further analysis could involve permutation feature importance (Altmann et al., 2010) of the decoding model trained on generated data to gain insights into learned representations.

4.1 Comparison with previous works

The Deep Recurrent Encoder (DRE) proposed by Chehab et al. (2022) is a highly relevant architecture to our approaches, as it demonstrates the advantages of modelling spatiotemporal dynamics for encoding neural data. DRE aims to predict MEG brain responses to visual word stimuli. While motivated as an encoding model, DRE can also be viewed through the lens of forecasting, with the addition of auxiliary task features. Forecasting holds inherent advantages over pure encoding, as it enables reconstructing real data and modelling complex spatiotemporal relationships, beyond just learning abstract representations.

Banville et al. (2021) investigate three SSL tasks for learning from unlabelled EEG recordings. Each task is trained via a contrastive loss function, where the model learns to pull positive pair examples closer in a representation space while pushing negative pairs apart. They demonstrate that the representations learned via SSL on unlabelled EEG data transfer well to supervised downstream tasks, consistently improving over limited label training and matching full supervision performance.

Building on this Kostas et al. (2021) propose combining self-supervised contrastive learning with Transformer networks to enable pre-training on large amounts of unlabelled EEG data. Their approach, BErt-inspired Neural Data Representations (BENDR), uses a Transformer encoder architecture applied to learned representations of raw EEG segments. First, a temporal convolutional network extracts initial representations of the EEG time series, referred to as BENDR features. Next, a Transformer encoder module takes the BENDR features as input. Contiguous segments of the BENDR representations are randomly masked, and the model is trained via a contrastive loss to predict the original features. Fine-tuning the pretrained model significantly improves performance on supervised EEG analysis tasks compared to training just on the downstream datasets.

In a similar vein, Wang et al. (2023) propose BrainBERT, a Transformer model tailored for self-supervised pretraining on unlabelled intracranial field potential recordings. BrainBERT is trained to reconstruct randomly masked patches of time-frequency spectrograms computed from the raw voltage measurements. Compared to BENDR, BrainBERT works directly on spectrogram representations rather than learned features. Similar to BENDR, the pretrained BrainBERT model serves as a general purpose feature extractor. By training linear classifiers on top of BrainBERT embeddings, Wang et al. (2023) achieve large performance gains on neural decoding tasks compared to raw or hand-engineered input features. Critically, BrainBERT generalises very well to new subjects and electrode locations.

In the image domain, tokenisation is often abandoned, and a linear projection directly maps image patches to continuous vector representations (Dosovitskiy et al., 2020). Similarly, Nie et al. (2022) have designed a channel-independent Transformer architecture applied to overlapping patches of continuous time series for forecasting. While this facilitates the input, without tokens categorical outputs cannot be generated. We posit that maintaining operations

over tokens and categorical outputs are desirable GPT2 features for M/EEG data. This is because we would like to output probability distributions and train using the cross-entropy loss. Previous applications of Transformer models to M/EEG data do not enforce a discrete tokenisation of the input, nor are they capable to generate new data auto-regressively. The latter allows for better interpretability of learned dynamics and artificial data simulation.

The tokenisation can happen either before or after mixing information across channels. The latter matches GPT2’s original design. One example of this is vector quantisation, which is used to tokenise multiple channels in Jukebox, a successful autoregressive Transformer model used on audio data (Dhariwal et al., 2020). Before training the Transformer, a hierarchical VQ-VAE (vector quantized variational autoencoder (Van Den Oord et al., 2017)) learns discrete codes (tokens) from raw audio. Once trained, VQ-VAE can map a continuous time series to a discrete token sequence \mathbf{z} . In the second step of Jukebox, the VQ-VAE is kept fixed, and the discrete tokens are used to learn an autoregressive Transformer.

Importantly, VQ-VAE is applied to single-channel audio to compress the temporal dimension into discrete codes. As our aim is to mix information across the channels we would primarily want to apply vector quantisation to the channel dimension, to have a discrete token at each timestep, or perhaps across a few timesteps. While an adaptation of this could work on MEG data, we experimented with a simpler non-deep learning method. We tried applying vector tokenisation on small groups of channels using the Residual Quantiser algorithm (Babenko and Lempitsky, 2014). Unfortunately, this approach to include information from all channels in the input sequence resulted in worse generation capabilities. See Supplementary Section A.1 for details on this method, and Supplementary Section A.3 and Section A.4 for results.

4.2 Limitations and future work

The full potential of self-supervised learning is only realised with large-scale data. This remains challenging for brain imaging compared to vision and language. Lowering barriers to data access and promoting data sharing is critical to realise the promise of foundation models in neuroimaging (Poldrack and Gorgolewski, 2014).

A core limitation of the channel-independent GPT2 model is no direct leveraging of cross-channel information for each sensor prediction. Our approach leveraging vector quantisation performed worse. We think that maintaining the innate inductive biases of Transformers, which emphasise 1D sequence modelling on embeddings of discrete tokens, is paramount. Different architectures or more data may enable proper utilisation of cross-channel dependencies. We tried various other approaches to mixing channel information beyond those reported, without success. For the Wavenet model, we incorporated all channels in the input by concatenating embeddings, and for the GPT2 models, we tried mixing channels with convolutions. We tried concatenating the output of each channel and then predicting from this shared output using a different projection for each channel. We also attempted to increase receptive field, dropout, and model size. One limitation in our approaches is the use of a next-timestep prediction loss. Future work should continue exploring architectures and different self-supervised or multi-timestep losses to leverage cross-channel information and improve modelling capabilities.

Some of our findings substantiated that predicting the next timestep may not serve as a robust measure of model performance. Future research should contemplate adopting multi-timestep

or contrastive loss frameworks. A plausible strategy could involve deploying the VQ-VAE model across both channel and temporal dimensions, aiming to distill a more coarse sequence of discrete tokens. Nevertheless, any quantisation-centric approach must carefully consider reconstruction error. We posit that a significant portion of the signal dynamics should be entrusted to the Transformer, given its adeptness in capturing complex dynamics.

A constraint in our modelling approach is its reliance on categorical task stimuli labels. Such an approach, while effective in our context, does not readily lend itself to scalability across diverse tasks and datasets. However, it is conceivable to construct robust representations tailored for various stimulus modalities—ranging from images to audio. These representations can then serve as conditioning embeddings. As shown by Défossez et al. (2022), tools such as wav2vec (Baevski et al., 2020) can be leveraged to derive informative representations of auditory stimuli.

Transfer learning also requires more thorough evaluation across diverse decoding tasks. It would be important to also investigate other more direct finetuning or transfer learning approaches of the forecasting model akin to vision or language domains. These could involve additional output layers and losses for finetuning on downstream tasks.

In conclusion, this work demonstrates that deep forecasting models can accurately reproduce complex neural dynamics of both ongoing and task-related activity and provides an extensive analysis methodology. Future work should explore more flexible conditioning, study different self-supervised and transfer learning frameworks, and critically, apply similar analyses when scaling up across diverse, large electrophysiology datasets. This has the potential to enable powerful transfer learning and advance foundational brain modelling and decoding.

Acknowledgments

This research was supported by the NIHR Oxford Health Biomedical Research Centre. The views expressed are those of the author(s) and not necessarily those of the NIHR or the Department of Health and Social Care. RC is supported by a Wellcome Centre Integrative Neuroimaging Studentship. MVE’s research is supported by the Wellcome Trust (215573/Z/19/Z). OPJ is supported by the UK MRC (MR/X00757X/1). MWW’s 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 and the Dementia Platform UK (RG94383/RG89702) and the EU-project euSNN (MSCA-ITN H2020-860563). The Wellcome Centre for Integrative Neuroimaging is supported by core funding from the Wellcome Trust (203139/Z/16/Z).

References

- Alammar, J. (2019). The illustrated gpt-2 (visualizing transformer language models). [Jalammar. github. io. https://jalammar.github.io/illustrated-gpt2](https://jalammar.github.io/illustrated-gpt2).
- Altmann, A., Tološi, L., Sander, O., and Lengauer, T. (2010). Permutation importance: a corrected feature importance measure. *Bioinformatics*, 26(10):1340–1347.
- Babenko, A. and Lempitsky, V. (2014). Additive quantization for extreme vector compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 931–938.

- Baevski, A., Zhou, Y., Mohamed, A., and Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.
- Baker, A. P., Brookes, M. J., Rezek, I. A., Smith, S. M., Behrens, T., Probert Smith, P. J., and Woolrich, M. (2014). Fast transient networks in spontaneous human brain activity. *elife*, 3:e01867.
- Banville, H., Chehab, O., Hyvärinen, A., Engemann, D.-A., and Gramfort, A. (2021). Uncovering the structure of clinical eeg signals with self-supervised learning. *Journal of Neural Engineering*, 18(4):046020.
- Beltagy, I., Peters, M. E., and Cohan, A. (2020). Longformer: The long-document transformer. [arXiv preprint arXiv:2004.05150](#).
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. [arXiv preprint arXiv:2005.14165](#).
- Chehab, O., Défossez, A., Loiseau, J.-C., Gramfort, A., and King, J.-R. (2022). Deep Recurrent Encoder: A scalable end-to-end network to model brain signals. *Neurons, Behavior, Data Analysis and Theory*, 1.
- Cichy, R. M., Khosla, A., Pantazis, D., Torralba, A., and Oliva, A. (2016). Comparison of deep neural networks to spatio-temporal cortical dynamics of human visual object recognition reveals hierarchical correspondence. *Scientific Reports*, 6(1):1–13.
- Csaky, R., van Es, M. W., Jones, O. P., and Woolrich, M. (2023a). Interpretable many-class decoding for meg. *NeuroImage*, 282:120396.
- Csaky, R., van Es, M. W. J., Jones, O. P., and Woolrich, M. (2023b). Group-level brain decoding with deep learning. *Human Brain Mapping*, 44(17):6105–6119.
- Cui, W., Jeong, W., Thölke, P., Medani, T., Jerbi, K., Joshi, A. A., and Leahy, R. M. (2023). Neuro-gpt: Developing a foundation model for eeg. [arXiv preprint arXiv:2311.03764](#).
- Défossez, A., Caucheteux, C., Rapin, J., Kabeli, O., and King, J.-R. (2022). Decoding speech from non-invasive brain recordings. [arXiv preprint arXiv:2208.12266](#).
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL*.
- Dhariwal, P., Jun, H., Payne, C., Kim, J. W., Radford, A., and Sutskever, I. (2020). Jukebox: A generative model for music. [arXiv preprint arXiv:2005.00341](#).
- 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](#).
- Fedus, W., Zoph, B., and Shazeer, N. (2022). Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *The Journal of Machine Learning Research*, 23(1):5232–5270.

- Gohil, C., Huang, R., Roberts, E., van Es, M. W., Quinn, A. J., Vidaurre, D., and Woolrich, M. W. (2023). osl-dynamics: A toolbox for modelling fast dynamic brain activity. [bioRxiv](#), pages 2023–08.
- Gohil, C., Roberts, E., Timms, R., Skates, A., Higgins, C., Quinn, A., Pervaiz, U., van Amersfoort, J., Notin, P., Gal, Y., et al. (2022). Mixtures of large-scale dynamic functional brain network modes. [NeuroImage](#), 263:119595.
- Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. [Journal of the royal statistical society. series c \(applied statistics\)](#), 28(1):100–108.
- Holtzman, A., Buys, J., Du, L., Forbes, M., and Choi, Y. (2020). The curious case of neural text degeneration. In [International Conference on Learning Representations](#).
- Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. (2020). Scaling laws for neural language models. [arXiv preprint arXiv:2001.08361](#).
- Kitaev, N., Kaiser, Ł., and Levskaya, A. (2020). Reformer: The efficient transformer. [arXiv preprint arXiv:2001.04451](#).
- Kostas, D., Aroca-Ouellette, S., and Rudzicz, F. (2021). Bendr: using transformers and a contrastive self-supervised learning task to learn from massive amounts of eeg data. [Frontiers in Human Neuroscience](#), page 253.
- Lewis, M. and MTSA, S. (1997). A-law and mu-law companding implementations using the tms320c54x. [Application Note SPRA163A](#), Texas Instrum., Dallas, TX, USA.
- Liu, S., Lu, H., and Shao, J. (2015). Improved residual vector quantization for high-dimensional approximate nearest neighbor search. [arXiv preprint arXiv:1509.05195](#).
- Meyes, R., Lu, M., de Puiseau, C. W., and Meisen, T. (2019). Ablation studies in artificial neural networks. [arXiv preprint arXiv:1901.08644](#).
- Nie, Y., Nguyen, N. H., Sinthong, P., and Kalagnanam, J. (2022). A time series is worth 64 words: Long-term forecasting with transformers. [arXiv preprint arXiv:2211.14730](#).
- Poldrack, R. A. and Gorgolewski, K. J. (2014). Making big data open: data sharing in neuroimaging. [Nature neuroscience](#), 17(11):1510–1517.
- Quinn, A. J., van Ede, F., Brookes, M. J., Heideman, S. G., Nowak, M., Seedat, Z. A., Vidaurre, D., Zich, C., Nobre, A. C., and Woolrich, M. W. (2019). Unpacking transient event dynamics in electrophysiological power spectra. [Brain topography](#), 32(6):1020–1034.
- Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. [Proceedings of the IEEE](#), 77(2):257–286.
- 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.
- Sutton, R. (2019). The bitter lesson. [Incomplete Ideas \(blog\)](#), 13(1).

- Torrey, L. and Shavlik, J. (2010). Transfer learning. In Handbook of research on machine learning applications and trends: algorithms, methods, and techniques, pages 242–264. IGI global.
- van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. (2016). WaveNet: A Generative Model for Raw Audio. In Proc. 9th ISCA Workshop on Speech Synthesis Workshop (SSW 9), page 125.
- Van Den Oord, A., Vinyals, O., et al. (2017). Neural discrete representation learning. Advances in neural information processing systems, 30.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, Advances in Neural Information Processing Systems 30, pages 5998–6008. Curran Associates, Inc.
- Vidaurre, D., Abeysuriya, R., Becker, R., Quinn, A. J., Alfaro-Almagro, F., Smith, S. M., and Woolrich, M. W. (2018a). Discovering dynamic brain networks from big data in rest and task. NeuroImage, 180:646–656.
- Vidaurre, D., Hunt, L. T., Quinn, A. J., Hunt, B. A., Brookes, M. J., Nobre, A. C., and Woolrich, M. W. (2018b). Spontaneous cortical activity transiently organises into frequency specific phase-coupling networks. Nature Communications, 9(1):1–13.
- Wang, C., Subramaniam, V., Yaari, A. U., Kreiman, G., Katz, B., Cases, I., and Barbu, A. (2023). Brainbert: Self-supervised representation learning for intracranial recordings. arXiv preprint arXiv:2302.14367.
- Wang, S., Li, B. Z., Khabsa, M., Fang, H., and Ma, H. (2020). Linformer: Self-attention with linear complexity. arXiv preprint arXiv:2006.04768.
- Wen, Q., Zhou, T., Zhang, C., Chen, W., Ma, Z., Yan, J., and Sun, L. (2022). Transformers in time series: A survey. arXiv preprint arXiv:2202.07125.
- Yuan, Z., Zhang, D., Chen, J., Gu, G., and Yang, Y. (2024). Brant-2: Foundation model for brain signals. arXiv preprint arXiv:2402.10251.
- Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., and Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. In Proceedings of the AAAI conference on artificial intelligence, volume 35, pages 11106–11115.

A Supplementary Material

A.1 FlatGPT2

FlatGPT2 is motivated by the observation that language models include extra information such as context within the sequence, instead of the feature space. We provide the detailed procedure below.

Directly vector quantising 300 channels to any vocabulary size would result in poor reconstruction. In FlatGPT2 we perform the tokenisation on small groups of channels instead. First, we compute the covariance over channels in the training data. Then, we apply K-means clustering (Hartigan and Wong, 1979) on the covariance matrix to group channels into buckets. This ensures that each bucket contains channels with high covariance. This is important because tokenising a feature space (group of channels) with high covariance can be done with fewer tokens while maintaining low reconstruction error. We set the number of clusters ($B = 30$) based on manual tuning on the training data. Each cluster/bucket can contain a variable number of channels, usually between 5 and 20.

After assigning channels to buckets we apply the Residual Quantiser algorithm (Babenko and Lempitsky, 2014) from the faiss library⁴ to each bucket b separately. This is a powerful additive quantiser (Liu et al., 2015) that achieves good reconstruction error with a relatively small vocabulary size V . Note that the total number of tokens, i.e. the vocabulary size will be BV , since we have B quantisers. Once fit to the training data the quantiser is fixed and can be applied to new data.

Mathematically, the covariance is obtained by:

$$\forall i, j \in 1, \dots, C \quad \mathbf{C}_{ij} = \frac{1}{T} \sum_{t=1}^T (x_{t,i} - \mu_i)(x_{t,j} - \mu_j) \quad (21)$$

Where $x_{t,i}$ is the i^{th} channel at timestep t , μ_i is the mean of channel i over all timesteps, and C is the total number of channels. \mathbf{C} is a symmetric matrix, and thus the feature and variable dimensions of K-means are the same. K-means computes buckets $\mathcal{C}_1, \dots, \mathcal{C}_B$ which partition channels C into distinct sets with high within-bucket covariance.

The residual quantiser Q_b learns a codebook $\mathbf{C}_b \in \mathbb{R}^{V \times |\mathcal{C}_b|}$ for each bucket \mathcal{C}_b :

$$\forall t \in 1, \dots, T \quad z_{t,b} = Q_b(\mathbf{x}_{t,b}; \mathbf{C}_b) \quad (22)$$

Where $z_{t,b}$ is the quantised representation (token/code) at timestep t of the channels $\mathbf{x}_{t,b} \in \mathbb{R}^{|\mathcal{C}_b|}$ in \mathcal{C}_b . The encoding in the quantiser is sequential, thus at stage m of the encoding of $\mathbf{x}_{t,b}$, the quantiser picks the entry i_m that best reconstructs the residual of $\mathbf{x}_{t,b}$ w.r.t. the previous encoding steps:

$$i_m = \operatorname{argmin}_j \|\mathbf{T}_m(j) - (\mathbf{x}_{t,b} - \mathbf{T}_1[i_1] + \dots + \mathbf{T}_{m-1}[i_{m-1}])\|^2 \quad (23)$$

⁴<https://github.com/facebookresearch/faiss/wiki/Additive-quantizers>

where \mathbf{T}_m is a table of size K_m containing $|\mathcal{C}_b|$ dimensional vectors. For notational simplicity we omit the index b from i_m and \mathbf{T}_m in the above. The quantisation provides a vector $[i_1, \dots, i_M]$, where each element i_m comes from a set of size $\lceil \log_2(K_m) \rceil$ bits. This bit vector representation can be easily transformed to token indices ranging from 1 to $V = \sum_{m=1}^M \lceil \log_2(K_m) \rceil$. Note that this table description of the discrete code is just a different representation of the overall codebook \mathbf{C}_b . A code $[i_1, \dots, i_M]$ can be reconstructed to obtain $\hat{\mathbf{x}}_{t,b}$ by retrieving the corresponding vectors from $(\mathbf{T}_1, \dots, \mathbf{T}_m)$ and adding them up. Reconstruction error is computed by comparing $\hat{\mathbf{x}}_t$ and \mathbf{x}_t .

By using 30 buckets we obtain 30 tokens per timestep, which is already a 10-fold reduction of the original dimension space, but we have not reached our initial goal of having 1 token per timestep. To achieve this, we flatten the feature dimension (buckets) when feeding tokens to GPT2, hence the name FlatGPT2. Our total sequence length then becomes $B \cdot T$, where B is the number of buckets and T is the number of timesteps. This approach is also motivated by the observation that language models include extra information such as context within the sequence, instead of the feature space. Thus, when predicting the token of bucket b , we treat the previous timesteps of the other buckets as contextual information.

We also add an extra separator token z_{sep} between sequences of buckets corresponding to the same timestep to facilitate distinction between the bucket and time dimensions. An input sequence to FlatGPT2 consists of tokens $z_{t,b}$ following a fixed order:

$$\mathbf{z} = (z_{sep}, z_{t=1,b=1}, z_{t=1,b=2}, \dots, z_{t=1,b=B}, \quad (24)$$

$$z_{sep}, z_{t=2,b=1}, z_{t=2,b=2}, \dots, z_{t=2,b=B}, \quad (25)$$

$$z_{sep}, \dots, z_{t=T,b=B}) \quad (26)$$

For each codebook \mathbf{C}_b a separate embedding $\mathbf{W}_{e,b} \in \mathbb{R}^{V \times E}$ is learned. As in ChannelGPT2 we add the appropriate conditioning embeddings to the input embedding with appropriate flattening across the channel/bucket dimension:

$$\mathbf{H}^{(0)} = \mathbf{Z}\mathbf{W}_e + \mathbf{W}_p + \mathbf{Y}\mathbf{W}_y + \mathbf{O}\mathbf{W}_o + \mathbf{W}_c + \mathbf{W}_t \quad (27)$$

where $+$ denotes element-wise addition and $\mathbf{Z} \in \mathbb{R}^{(B+1)T \times V}$ is the one-hot version of \mathbf{z} . The task labels \mathbf{Y} can vary across time, but are the same across the buckets of one timepoint. \mathbf{W}_c now contains distinct embeddings of buckets $b \in (1, \dots, B)$, which are the same across timesteps. We also augment the input with \mathbf{W}_t , containing distinct embeddings for timesteps $t \in (1, \dots, T)$, which are the same across buckets. This is the timestep version of \mathbf{W}_c .

As usual, the model is trained to autoregressively predict the next token in the sequence given all previous inputs. At timestep t and bucket b the model has access to the tokens $\mathbf{z}_{1:t-1}$ from all buckets (and thus information from all channels), and the tokens $\mathbf{z}_{t,1:b}$, and has to predict token $z_{t,b+1}$. The buckets of the same timestep are predicted sequentially, thus, bucket ordering could influence results. We use an arbitrary bucket ordering and do not experiment with different orderings of the input sequence.

Note that at the last bucket B in each timestep the prediction should be token z_{sep} , however, we simply discard this prediction during loss computation, as we do not require the model to

Description	Parameter	Typical value
Number of buckets	B	30
Number of code tables	M	2
Number of bits per code table	$\lceil \log_2(K_m) \rceil$	7
Vocabulary size per bucket	$V = \sum_{m=1}^M K_m$	16384

Table 4: Hyperparameters of the vector quantisation part of FlatGPT2.

predict separator tokens. The structure of the sequence already constrains the predictions such that a new timestep begins after every B tokens. Conversely, when computing the prediction at input token z_{sep} , the target is the token with bucket $b = 1$ of the next timestep. This is useful as in theory we could start the recursive generation of data with a single z_{sep} token.

At the output, the transpose of \mathbf{W}_e can be used to predict probabilities over the vocabulary, or a separate linear projection can be learned. Note that because each codebook \mathbf{C}_b has a separate vocabulary of size V assigned to it, we can speed up the output softmax by only computing probabilities over codes/tokens in \mathbf{C}_b instead of the total vocabulary of size BV .

FlatGPT2 contains important hyperparameters that affect design choices and performance (Table 4). Increasing the number of buckets B improves reconstruction error, as the vector quantiser has to quantise less channels, but increases the length of the input sequence to FlatGPT2, and the total size of the vocabulary BV . The number of code tables M and the number of bits per code table define the size of the vocabulary $V = \sum_{m=1}^M K_m$. These were manually tuned, but generally, we observed that using fewer code tables with a higher number of bits achieves lower reconstruction error. For example, a vocabulary size of 16 bits can be achieved with both two 8-bit code tables and four 4-bit code tables. The trade-off is that using fewer code tables (with more bits) significantly increases computation time. Increasing the vocabulary V (through the number of code tables and bits per table) improves reconstruction error, as more codes are available for quantising a bucket of channels. However, this increases the total vocabulary BV of the model, resulting in a larger model.

In summary, key modifications compared to ChannelGPT2 include vector quantisation (tokenisation) of channel groups, and flattening the channel dimension into the sequence. While in theory we could have flattened the full channel dimension without bucketing, this would have resulted in a 10x longer sequence length. However, we are limited by memory constraints since a standard GPT2 model scales quadratically with the sequence length. Memory-efficient Transformer variants are an active research area (Kitaev et al., 2020; Beltagy et al., 2020; Wang et al., 2020), but they have other drawbacks, and we leave their application to M/EEG data to future work.

For FlatGPT2 we set the (temporal) receptive field to be between 120 and 240 because of memory constraints. Note that the total (actual) receptive field of the model is the temporal receptive field multiplied by the number of buckets + 1. All embedding sizes were set to 96, and we used 8 GPT2 layers, with 8 attention heads. Dropout was set to 0 and we used early stopping on the validation set. The quantisation parameters are given in Table 4.

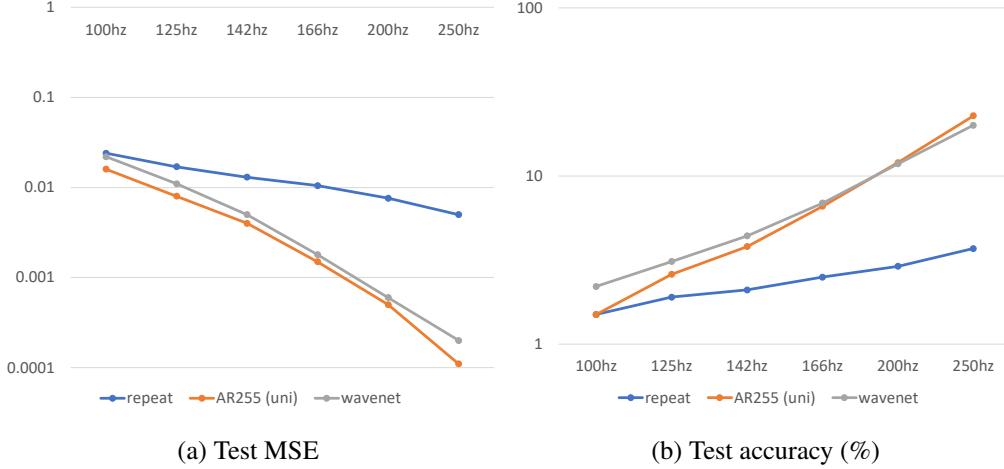


Figure 18: Comparing AR(255) and WavenetFullChannelMix (wavenet) across increasing sampling rates of the data. *repeat* refers to the repeat baseline. Accuracy is given in percentages.

A.2 Effect of sampling rate on forecasting performance

We further analysed sampling rate effects on forecasting performance in Figure 18. We trained the AR(255) and WavenetFullChannelMix models on increasing sampling rates of the data from 100 Hz to 250 Hz. The lowpass filter was kept the same at 50 Hz. The receptive fields were kept the same in terms of timesteps, thus they decreased accordingly in terms of actual time in seconds. As expected, both AR and Wavenet models improved markedly with higher sampling rates, as the prediction task became easier when timesteps were closer together. The performance gap between models and the repeating baseline also grew with sampling rate. However, these trends are likely influenced by both the changing prediction interval and filtering artefacts. It is unlikely that such marked improvement would be caused by better modelling of higher-frequency content. Varying the low-pass cut-off with sampling rate reduced performance, suggesting filtering effects dominate. Removal of noise with lower lowpass filters is also a possible explanation.

A.3 Generated covariance

On most metrics presented in the main paper FlatGPT2 performs considerably worse than ChannelGPT2. As the PSD is a channel-independent measure, we also looked at generated data covariance which captures the interactions between different channels (Figure 19). This reveals that the only model capable of closely matching the data covariance is FlatGPT2. All other models produce data with covariances much closer to 0. This is perhaps expected for channel-independent models which generate data independently for each channel, but somewhat surprising for WavenetFullChannelMix. Even though FlatGPT2 may not produce accurate spectral data, by having information about other channels in the input it does an excellent job at capturing covariance. This highlights the trade-offs between different model architectures.

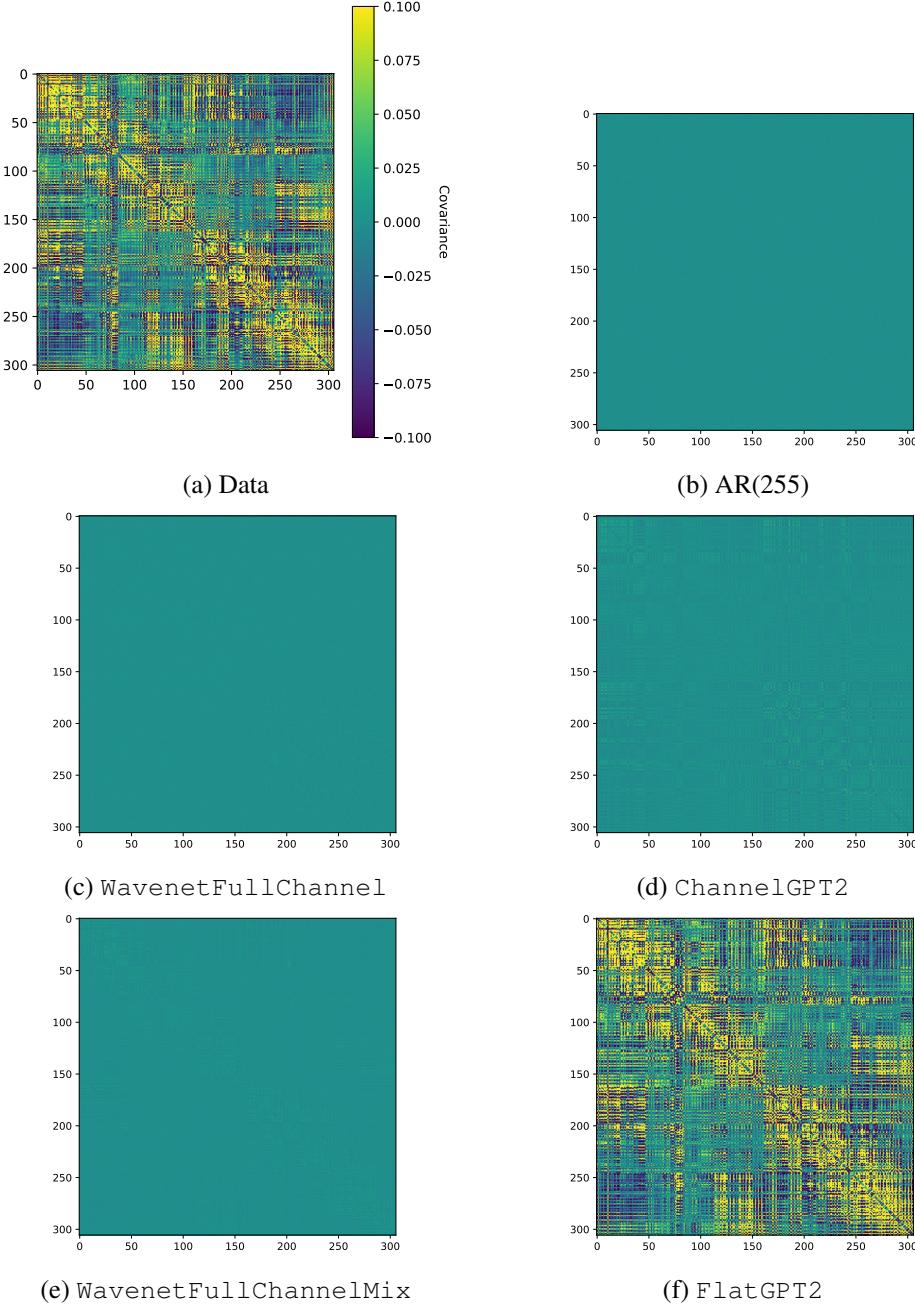


Figure 19: Covariance of generated data between channels (vertical and horizontal axes). All plots have the same scaling as (a).

A.4 FlatGPT2 on group data

Unfortunately, even scaling FlatGPT2 did not improve evoked generation. However, we did find that the spectral content of the generated data matched the real data much better than the single-subject version of FlatGPT2 (Figure 20). FlatGPT2-group seemed to scale particularly well with model size as larger models achieved lower and lower loss, improving test accuracy by multiple folds (16.1% top-1 and 40.1% top-5 accuracy) over single-subject

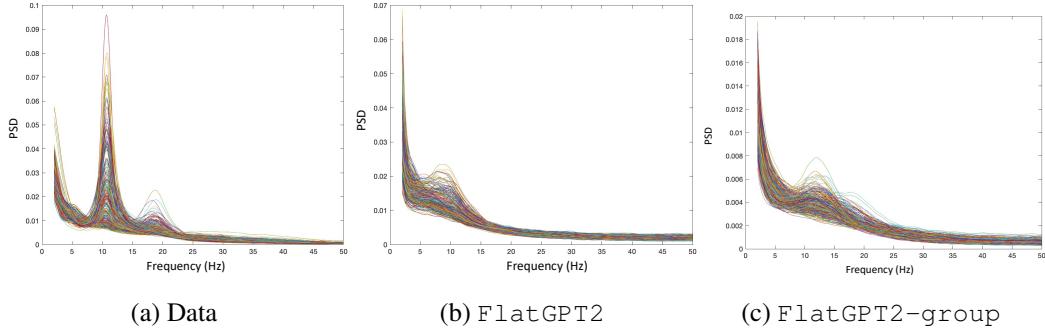


Figure 20: Comparison of generated data PSD across data single-subject FlatGPT2 and FlatGPT2-group. Each line represents a different MEG channel.

FlatGPT2 (3% top-1 and 10.8% top-5 accuracy). This is interesting behaviour compared to ChannelGPT2-group which did not improve much on our forecasting metrics. It remains to be seen whether even more data and larger models are needed to make this type of architecture viable.

LLMChain: Blockchain-based Reputation System for Sharing and Evaluating Large Language Models

Mouhamed Amine Bouchiha, Quentin Telnoff, Souhail Bakkali, Ronan Champagnat, Mourad Rabah,

Mickaël Coustaty, Yacine Ghamri-Doudane

L3i - La Rochelle University, La Rochelle, France

{mouhamed.bouchiha, quentin.telnoff, souhail.bakkali, ronan.champagnat, mourad.rabah, mickael.coustaty, yacine.ghamri}@univ-lr.fr

Abstract—Large Language Models (LLMs) have witnessed a rapid growth in emerging challenges and capabilities of language understanding, generation, and reasoning. Despite their remarkable performance in natural language processing-based applications, LLMs are susceptible to undesirable and erratic behaviors, including hallucinations, unreliable reasoning, and the generation of harmful content. These flawed behaviors undermine trust in LLMs and pose significant hurdles to their adoption in real-world applications, such as legal assistance and medical diagnosis, where precision, reliability, and ethical considerations are paramount. These could also lead to user dissatisfaction, which is currently inadequately assessed and captured. Therefore, to effectively and transparently assess users' satisfaction and trust in their interactions with LLMs, we design and develop LLMChain, a decentralized blockchain-based reputation system that combines automatic evaluation with human feedback to assign contextual reputation scores that accurately reflect LLM's behavior. LLMChain helps users and entities identify the most trustworthy LLM for their specific needs and provides LLM developers with valuable information to refine and improve their models. To our knowledge, this is the first time that a blockchain-based distributed framework for sharing and evaluating LLMs has been introduced. Implemented using emerging tools, LLMChain is evaluated across two benchmark datasets, showcasing its effectiveness and scalability in assessing seven different LLMs.

Index Terms—Blockchain, LLMs, Decentralized Reputation, Transparency, Human Feedback, Automatic Evaluation.

Paper accepted at IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC) IEEE, Osaka, Japan (2024).

I. INTRODUCTION

LARGE Language Models (LLMs) have received a great deal of attention in the last few years due to their surprising capabilities in managing a wide range of Natural Language Processing (NLP) tasks including information retrieval, language understanding, generation, and reasoning [1], [2]. Despite their impressive capabilities, LLMs such as GPT-3, Llama, and Vicuna [3]–[5] exhibit certain challenges that compromise their efficacy. One prominent issue is the manifestation of biases and fairness concerns. LLMs often inherit biases present in their training data, reflecting societal prejudices and stereotypes [6]. Consequently, these models can produce outputs that perpetuate or even exacerbate existing social inequalities. Another limitation arises from the models' difficulty in grasping common sense and contextual understanding. LLMs may struggle to interpret nuances in

language, leading to responses that appear nonsensical or detached from real-world knowledge [7]. These behaviors encompass hallucinations, evident in the generation of text that invents or imagines information lacking a factual or coherent basis [8]. LLMs may also display unreliable reasoning [9], characterized by a lack of consistent or dependable logical abilities. Furthermore, there is a risk of generating harmful content [10], where LLMs may produce material that is offensive, inappropriate, or potentially harmful. These behaviors can significantly deviate from the expected or desired output, undermining the credibility of LLMs and posing challenges to their widespread adoption. In summary, these flawed actions that diminish trust in LLMs cause users to be cautious about relying on AI-generated content due to its unpredictability and potential for producing incorrect information. They also present hurdles to the utilization of LLMs in critical contexts such as medical diagnosis, legal advice, or sensitive information processing, where accuracy and reliability are essential.

One key way to assess the behavior of LLMs and measure their reliability involves soliciting inputs from users. Individuals can highlight issues they encounter while engaging with AI-generated content [11]. However, this method has two notable drawbacks. First, collecting user feedback is costly as it requires analyzing and categorizing the gathered information. Second, human feedback lacks real-time capabilities as users might not offer immediate responses. This delay hinders prompt evaluation given the absence of instant responses from humans. Therefore, to reduce reliance on human involvement, an alternative strategy consists of employing automatic evaluation methods. These techniques leverage automated feedback [2], [10] or language models [12], [13] to evaluate LLMs' performance in a cost-effective way. Despite the efficient processing of language data generated by LLMs, the automatic evaluation metrics they rely on may not perfectly align with human preferences or perceptions, thereby introducing certain limitations. These assessments may fail to capture nuances or qualitative aspects that are crucial for understanding how users perceive the content generated by LLMs [14]. Additionally, existing human and automatic evaluation-based methods face many challenges linked to the lack of transparency and decentralization, as they currently all operate within centralized frameworks. Entities wishing to use LLMs for specific tasks must choose between trusting centralized third-party evaluations or independent testing, which is a costly process that

depends on the availability of code and data. Moreover, most of the recent studies concentrate on either human feedback or automated evaluation [10], [11], [15], [16], missing the opportunity to capture human preferences while enhancing scalability and reducing costs.

To address the above-mentioned issues of evaluating LLMs effectively, dynamically, and transparently, we propose LLMChain, which leverages Blockchain (BC) technology to build a reputation system for LLMs. Blockchains have found extensive use in various trust-related applications such as supply chain [17], crowdsourcing [18], and e-commerce platforms [19]. Its utilization is particularly essential for the development of efficient, decentralized, and transparent reputation systems. These attributes are precisely the qualities we have always envisioned for developing robust reputation systems. Blockchain - known for its resistance to tampering - can be used to track and manage the reputation of various LLMs via smart contracts. LLMChain's primary goal is to help users find the most reliable LLM that meets their specific needs and preferences. Therefore, it allows these individuals to use language models shared by LLM providers and actively participate in the evaluation process. Additionally, it provides LLM developers with valuable insights, enabling them to enhance and optimize their models by incorporating human feedback. Besides, it is discouraged within reputable organizations for employees to disseminate professional data online or to external entities, a practice that is frequently observed with commercial LLMs. LLMChain aims to address this issue by enabling these organizations to identify open-source LLMs that meet their needs and capabilities for local deployment. This privacy assurance also extends to users who prefer not to share their activities and personal data with third parties. In summary, the contributions of this paper are:

- A new reputation-based model. This one is proposed to assess user satisfaction and determine the level of trust associated with each interaction with a language model, via a comprehensive yet scalable evaluation of LLMs' responses (using human feedback and automatic evaluation).
- A fully decentralized, blockchain-powered platform that enables LLMs to be shared and evaluated thanks to the designed reputation-based model.
- The preparation of LLMGooAQ¹, a comprehensive dataset encompassing diverse questions and answers across various domains and contexts. This dataset consists of over 100k questions pulled from the large-scale GooAQ dataset and their corresponding answers obtained by performing inference on seven open-source LLMs.
- An extensive experimental evaluation with multiple scenarios is performed to demonstrate the effectiveness of the proposed reputation model and the scalability of LLMChain.

II. RELATED WORK

A. LLMs Evaluation

To assess the credibility and capabilities of LLMs, several studies have introduced diverse evaluation methods, including pairwise comparison, single-answer grading, or reference-guided grading, employing another LLM as an evaluator. [2], [15]. These methodologies offer advantages in scalability and interoperability. Nevertheless, it comes with notable limitations: 1) Position Bias, where the evaluator tends to favor the initial model; 2) Verbosity Bias, where the evaluator prefers longer responses over shorter ones; and 3) Self-Enhancement/Promotion Bias, where the judging model prioritizes its own text or that generated from a similar model. Moreover, evaluating a LLM using another LLM appears paradoxical since the evaluator itself is subject to evaluation. On the other hand, alignment-based methods are used to make large-scale alignment research more accessible like OpenAssistant conversations [20], which is a corpus of conversations that resemble interactions with assistants, created and annotated by humans. Nonetheless, alignment-based methods face some scalability challenges and annotation expenses. In Core-GPT [21] and [22], authors focus on assessing the credibility of LLMs. Core-GPT [21] proposes an approach that combines open-access scientific literature with LLMs to improve their reliability and trustworthiness. However, its methodology's scope is limited to two LLMs, "GPT3.5" and "GPT-4", failing to illuminate the credibility gap between open-source and commercial models. In contrast, the approach proposed in [22] introduces an automated workflow designed to manage an increased number of requests/responses, facilitating the assessment of the credibility of multiple LLMs. In G-Eval [16], which is a framework that leverages large language models, used a Chain-of-Thoughts (CoT) and a form-filling paradigm to evaluate the quality of Natural Language Generation (NLG) outputs. G-Eval experimentation involves two generation tasks: text summarization and dialogue generation. However, here again, the methodology is limited to only two LLMs which are "GPT3.5" and "GPT-4".

When delineating the prevailing approaches employed to assess the credibility of LLMs, typical challenges become apparent. These approaches lack transparency and decentralization as they all operate within centralized frameworks. To determine the most credible LLM for a specific context, individuals are faced with two alternatives: either relying on centralized evaluations or carrying out tests independently. Additionally, the majority of current studies focus on either human feedback or automated evaluation separately, missing an opportunity to effectively capture human preferences while enhancing scalability and reducing costs.

B. Blockchain-based Reputation Systems

The inherent decentralized and tamper-proof nature of blockchain technology provides essential attributes for effective reputation management. Several blockchain-based reputation systems exist, demonstrating the maturity and usability

¹<https://github.com/mohaminemed/LLMGooAQ/>

of such solutions for novel applications. TrustChain [17] is a three-layered blockchain-powered framework used for trust management in IoT-supported Supply Chains. The solution constitutes a service platform operating on a permissioned blockchain network. It leverages smart contracts to automate the computation of reputations and incorporates an incentive mechanism based on rewards and penalties to motivate users toward proper behavior. GuRuChain [19], introduces a blockchain-based service trading platform that incorporates guarantee and reputation at application and consensus layers to foster accountability and trust. It leverages smart contracts to implement the proposed reputation model and manage guarantees using deposits. ValidatorRep [23], is a verification scheme that utilizes blockchain with trust management to foster accountability within crowdsourcing systems. Specifically, this proposal entails a decoupled blockchain model designed for the distinct storage of business transactions and log transactions throughout data interaction. It uses a trust model encompassing the reputation of participants and the trust relationships among them. In REPUTABLE [24], the authors propose a decentralized reputation system for assessing service providers' activity within a blockchain-based ecosystem. The proposed solution integrates a centralized oracle to perform off-chain computations and triggers on-chain smart contracts, impeding the system from achieving complete decentralization. TRUSTD [25] is an ecosystem powered by blockchain and collective signatures, designed to support content creators in garnering community backing for their content. Additionally, it aids users in assessing the credibility and accuracy of these contents.

Therefore, to address the aforementioned challenges related to LLM's evaluation, we believe in the consistency of extending the use of such reputation systems, proposing a novel decentralized framework for evaluating LLMs on open-ended question answering. The proposed concept aims to build a robust and transparent blockchain-based reputation system that merges human evaluation with automated metrics to assess LLMs responses effectively. To our knowledge, this work represents the first study of language model evaluation in a decentralized setting.

III. LLMCHAIN FRAMEWORK

In this section, we introduce LLMChain, a Blockchain-powered reputation system for LLM's evaluation. In particular, the proposed framework aims to foster trust in LLMs by amalgamating human feedback and automated evaluations. LLMChain can be seen as a decentralized reputation-based store that allows sharing and evaluating LLMs. It serves a dual role by addressing the needs of users seeking reliable AI assistance, as well as assisting LLMs developers in enhancing the performance and reliability of their models. Fig.1 illustrates an overview of the proposed LLMChain framework.

A. LLMChain Architecture

The proposed LLMChain framework is composed of multiple entities distributed over four main layers as depicted in

Fig.1a.

- **User Layer:** is composed of individual participants. Each participant has at least one end-device to interact with the system. Users with different areas of expertise can join the system to use shared, open-access LLMs and provide feedback after engaging with any of the models. This allows users not only to gain insights into the most suitable LLM for their specific domains but also to actively participate in the evaluation process by testing these models and sharing their feedback.
- **Blockchain Layer:** functions as a permissioned blockchain, comprising nodes initiated by LLM providers and/or developers. To participate in the network, an entity must develop and share at least one LLM. LLMChain network employs a consensus mechanism to uphold a uniform ledger copy. We advocate for a reputation-based consensus [19], [26], leveraging an existing reputation model within the system. Compared to traditional consensus methods, reputation-based consensus offers scalability and enhanced fairness. To further improve the accessibility and performance of our decentralized application, we introduce an InterPlanetary File System (IPFS) [27] as an off-chain storage system. The core business logic of LLMChain is securely executed via smart contracts deployed over the network and accessed through the submission of transactions. LLM providers benefit from joining the network by gaining full access to LLMChain and, consequently, all the evaluations occurring within the system. This access allows them to accumulate extensive information that will help them to improve and correct their models.
- **Oracle Layer:** comprises Oracle nodes that merge on-chain code with off-chain infrastructure, facilitating the creation of a sophisticated Decentralized Application (DApp). This application responds to real-world events and seamlessly interacts with conventional systems (LLM servers). Hybrid smart contracts deployed across the decentralized Oracle network enable automating the evaluation process. The network intercepts responses from models, conducts off-chain automatic evaluations, and subsequently triggers on-chain smart contracts to update the overall score of the targeted LLM. All of that is achieved in a decentralized and trustless way through the execution of an Oracle protocol [28].
- **LLM Layer:** consists of language models that are administered locally by LLM providers and/or developers. For users who wish to utilize these models for inference tasks, developers need to maintain ongoing access to their shared models. The Oracle network conducts regular checks on the connectivity of these shared models. Any model that goes offline is automatically removed from the list of running models, keeping the view up-to-date and avoiding interaction with non-operating models.

B. LLMs' Evaluation process in LLMChain: An overview

Unlike centralized frameworks where the evaluation is implemented by a third party, we define end-to-end decentralized evaluation protocols. The proposed protocols are implemented

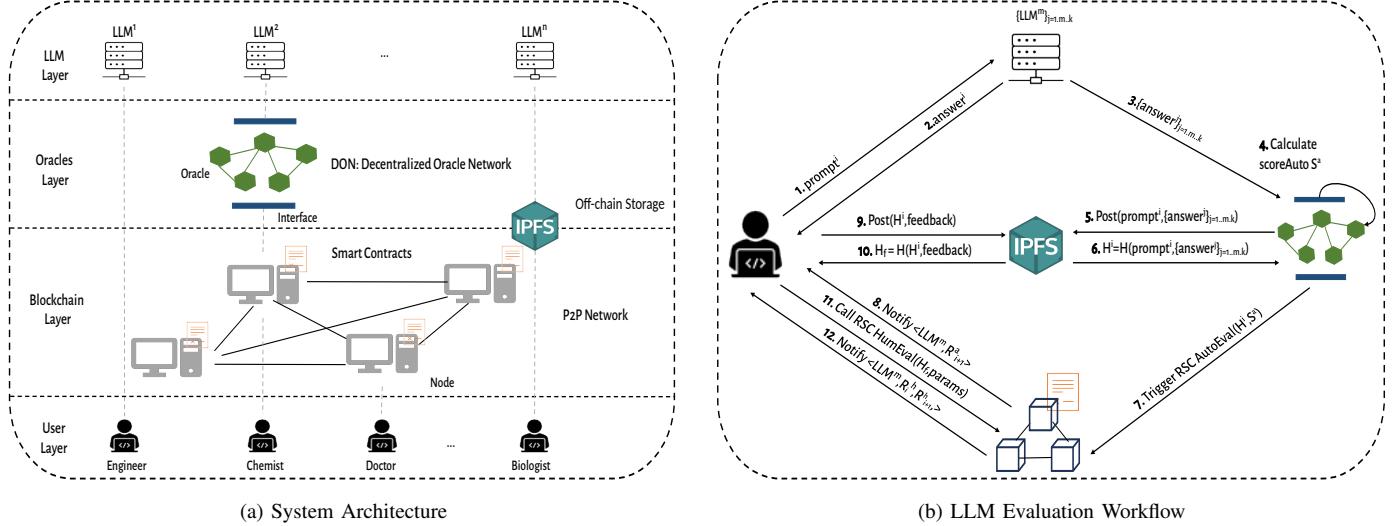


Fig. 1: Overview of the LLMChain framework. 1a presents the layered BC-powered architecture. It consists of four main layers: a user layer formed by individuals with different expertise, a BC layer built on a consortium BC managed by LLM providers, and an Oracle layer built up by a decentralized network interconnecting the BC layer with LLMs layer. 1b describes the LLMs evaluation process in LLMChain.

in the LLMChain architecture using smart contracts. The evaluation process consists of three main phases:

1) *Registration*: To obtain their credentials, including public (address) key and private key, *Users* and *Developers* must register on LLMChain through the Identity Smart Contract (ISC). The registration process can be done in a decentralized, privacy-preserving, and Sybil-resistant way using an IDentity Management Ledger (IDML) [29].

2) *LLM Sharing*: LLM developers can add a new model to LLMChain via Reputation Smart Contract (RSC) by calling the *addModel* function. This creates a new $LLM = \{CID_{llm}, Owner, R_0^a, R_0^h, R_0\}$. *Owner* is the developer's public key. The initial human R_0^h , automatic R_0^a , and weighted reputations R_0 for the model are calculated as the average values across all existing models in the system. CID_{llm} is the hash of the model's details published on IPFS (*i.e.* The Content Identifier). To ensure the security of LLMChain's smart contract functionalities, we implement role-based access control to manage permissions. This is realized through the Access Control Smart Contract (ACSC). ACSC restricts calling functions by role, for example, it restricts the ability to share models on LLMChain to developers only.

3) *LLM Evaluation*: The comprehensive process, spanning from prompt submission to updating the global reputation for the chosen model is illustrated in Fig.1b. It begins with the user formulating a request intended for a specific LLM_m , directly transmitted to the model via a dedicated interface (API). Subsequently, the response from LLM_m is relayed back to the user. To perform **Automatic evaluation**, the Oracle intercepts both the request and the response. Then, it dispatches identical prompts to other k models $\{LLM_j\}_{j=1,\dots,k}$, to use their answers as comparative references. Next, it calculates the automatic score for LLM_m using the model described in Sec. IV-B1. After that, it stores the prompt and its corresponding

answers off-chain using IPFS. Finally, it triggers the RSC to update the overall automatic score of LLM_m by calling the *autoEval* function. Upon receiving the answer, users can opt for direct **Human evaluation** by calling the *humEval* function or seek alternative candidate responses to gauge the quality of LLM_m 's answer *i.e.* using the shared hash H^i , they can retrieve all k answers from IPFS. Once this operation has been completed, the overall weighted reputation score is updated by calling the *updateReputation* function. Further details on the automatic and human evaluation procedures follow in the next section.

IV. REPUTATION MODEL

Human evaluation entails the participation of human experts or users to assess the quality, coherence, and overall adequacy of generated answers. These metrics seek to encompass subjective aspects that automated metrics may overlook [14]. Nevertheless, evaluating generated answers through human feedback poses challenges as it relies on users' willingness to offer genuine and immediate feedback. To better address these, we investigate automatic methods, enabling LLMChain to evolve even in the absence of human feedback. In this section, we introduce our reputation model that blends human and automated evaluations. This approach aims to leverage the efficiency and scalability of automated methods while upholding strong alignment through human feedback.

A. Reputation Formulation

We model the reputation of an LLM as a tuple denoted by $REP = \{R^a, R^h, R\}$. Our approach involves assigning an initial reputation, noted $REP_0 = \{R_0^a, R_0^h, R_0\}$, to each new LLM. The values of R_0^a , R_0^h , and R_0 are derived from the average scores of all LLMs in the system.

The REP tuple undergoes updates after each interaction i , following two stages: i) **Interaction Evaluation**, which

involves computing three scores for the targeted LLM - an automatic score S^a , a human score S^h , and a weighted combination S^θ between both scores - with their respective weights ω^a , ω^h , and ω^θ . ii) **Global Scores Updating.** each global score R_i in REP is updated using a predefined function securely implemented in the RSC contract. For each $(R_i, S_{calc}, \omega) \in \{(R^a, S^a, \omega^a), (R^h, S^h, \omega^h), (R, S^\theta, \omega^\theta)\}$,

$$\begin{aligned} \mathcal{U} : [0, 1] \times [0, 1] \times [0, 1] &\longrightarrow [0, 1] \\ (R_i, S_{calc}, \omega) &\longrightarrow R_{i+1} \end{aligned} \quad (1)$$

B. Interaction Evaluation

1) *Automatic Evaluation:* Several studies have demonstrated that embedding-based metrics can effectively match human judgments by considering semantic relevance [30], [31]. However, their effectiveness is influenced by the quality of the underlying embedding. Consequently, when developing LLMChain, we emphasized a modular framework to retain flexibility in updating the automatic evaluation technique at any time. The metrics we explore to use for the **Automatic evaluation** requires a minimum of one reference to compute the score S^a (cf. Sec. V-A3). Thus, we propose to use k references, denoted as $\{ref^j\}_{j=1\dots k}$ to evaluate the answer of the targeted model for better precision. The k references are the answers that the decentralized Oracle gets from the top k models within the context of the prompt. The final score of the answer from the model LLM is computed as follows:

$$S^a = \frac{1}{k} \sum_{j=1}^k scoreAuto(answer, ref^j) \quad (2)$$

We assess the quality of the automatic evaluation using a weighting function $\omega^a \in [0, 1]$. Its outcome varies depending on the average reputation of the models used as references (*i.e.* the better the reputation the higher importance is given). Once this is done, the Oracle triggers the *autoEval* function in RSC to update the overall automatic score of the LLM_m using the model described in the Sec. IV-C.

2) *Human Evaluation:* While it is straightforward to carry out an automated evaluation by measuring the distance/similarity between generated answers, it is less easy to gather information about trust, user satisfaction, completeness, and usefulness of a generated text. Inspired by [14] and [32], our approach involves employing a multi-item scale questionnaire for efficient and scalable human evaluation. Our focus encompasses two types of dimensions (constructs) essential for users to assess text generated by LLM accurately:

- **Answer's Constructs:** are the metrics that allow the evaluation of the quality of a single answer/response (*i.e.* calculate S^h). To do so, we employ three metrics. First, the **Reliability**, denoted as A_t , evaluates the trustworthiness of the provided answer. Then, the **Completeness**, denoted as A_c , measures the comprehensiveness or completeness of the answer. Finally, the **Utility**, denoted as A_u , determines the

usefulness of the answer. The human score of an answer is a linear combination of the three metrics:

$$S^h = [\alpha_r A_t + \beta_r A_c + \gamma_r A_u]; \alpha_r + \beta_r + \gamma_r = 1 \quad (3)$$

- **User Constructs:** encompass parameters that signify a user's proficiency and ability in evaluating the generated text, showcasing the quality of their assessment and its influence on the overall human score (*i.e.* calculate ω^h). To do so, we define four metrics. First, **Duration**, denoted as D , measures the time interval in minutes between the last two evaluations. Second **Familiarity**, denoted as F , gauges the user's familiarity with the response context. Third, **LLM Trust**, denoted as T , assesses the user's belief in the expertise of the targeted LLM. Finally, **Uncertainty**, denoted as U , captures the user's degree of uncertainty regarding the evaluation. The weight of the human evaluation is given by:

$$\omega^h = \mathcal{W}^h \mathcal{F}_D \quad (4)$$

Where,

$$\mathcal{W}^h = [\alpha_u F + \beta_u T + \gamma_u (1 - U)]; \alpha_u + \beta_u + \gamma_u = 1$$

and,

$$\mathcal{F}_D = \tanh_\lambda(D) = \frac{1 - e^{-\lambda \cdot D}}{1 + e^{-\lambda \cdot D}}$$

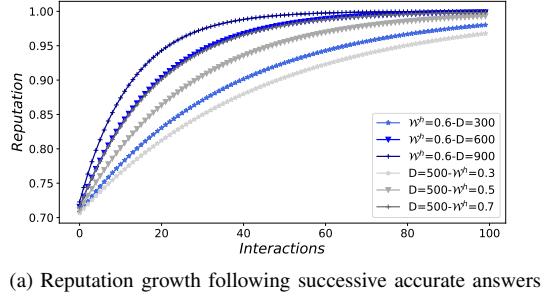
We normalize D using a hyperbolic tangent function $\mathcal{F}_D \in [0, 1]$. \mathcal{F}_D is implemented in a way that thwarts potential abuse. It reduces the impact of successive evaluations performed within a short period, thereby protecting the LLM's overall reputation and reinforcing the model's effectiveness. Furthermore, the positive correlation with the other metrics (*i.e.* F , T , and $1-U$) leads to important considerations: first, ratings from users less familiar with the context carry less weight in updating the model's overall human reputation; second, ratings from users with minimal trust or with higher uncertainty have less impact on updates compared to those with lower uncertainty and higher trust in the overall expertise of LLMs.

C. Overall Scores Update

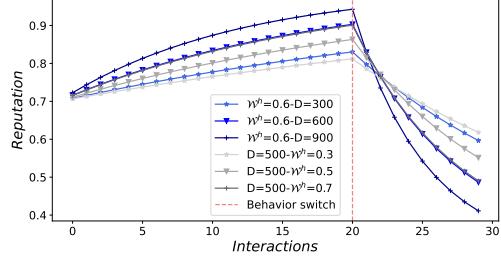
In LLMChain, we employ three types of updates. The overall automatic reputation R^a update occurs after each interaction to keep tracking the LLM behavior, while changes in R^h and R only occur if the interaction includes a human evaluation. These updates depend on the outcome of the automatic evaluation S^a , the human evaluation S^h , or the weighted evaluation S^θ . We use θ , a configurable weighting factor, to give more emphasis to the human evaluation when calculating S^θ and ω^θ , as follows:

$$\begin{cases} S^\theta = \theta S^h + (1 - \theta) S^a \\ \omega^\theta = \theta \omega^h + (1 - \theta) \omega^a \end{cases} \quad (5)$$

The updating formula $\mathcal{U}_{\psi, \xi} : (R_i, S_{calc}, \omega) \longrightarrow R_{i+1}$ for the three scores R^h , R^a , and R is thus defined as follows:



(a) Reputation growth following successive accurate answers



(b) Reputation changes after successive incorrect answers

Fig. 2: The Effectiveness of LLMChain’s Reputation model under different \mathcal{W}^h and D .

$$\forall (R_i, S_{calc}, \omega) \in \{(R^a, S^a, \omega^a), (R^h, S^h, \omega^h), (R, S^\theta, \omega^\theta)\},$$

$$R_{i+1} = \begin{cases} (1 - \psi\omega)R_i + \psi\omega S_{calc}; & S_{calc} \geq \bar{R}_i \\ (1 - \xi\omega)R_i + \xi\omega S_{calc}; & S_{calc} < \bar{R}_i \end{cases} \quad (6)$$

where R_i and \bar{R}_i are the current reputations and trust thresholds (*i.e.* before the interaction i), respectively. We define the threshold \bar{R}_i as the average of LLM reputations.

By employing two distinct formulas in (Eq. 6) for the update process using a trust threshold \bar{R}_i , we separate expected good behavior from unexpected bad behavior (no/bad response, hallucination, harmful content, etc. [33], [34]). Consequently, we can put more weight (*i.e.* $\xi > \psi$) on the newly calculated score S_{calc} in the case of an incorrect response. Moreover, the integration of the weighting function ω into both equations establishes a direct relationship between the quality of the evaluation and its impact on the update of the overall reputation. For instance, for a R^h update, the greater the user’s familiarity, certainty, and trust in the LLM expertise, the more significant their evaluation’s impact becomes. Moreover, the use of D allows the system to mitigate consecutive inaccurate ratings that may be intended to enhance or damage LLM’s reputation. We note that this metric is reset at regular intervals (*e.g.* every 24 hours), preventing users who abstain from evaluations for a long time from exploiting the model.

Fig. 2 demonstrates the impacts of D and \mathcal{W}^h on the overall reputation updates. It shows the shifts in reputation between a skilled model consistently providing accurate responses and a less competent one that produces consecutive incorrect answers after delivering multiple correct ones. Both positive and negative updates have a direct correlation with D and \mathcal{W}^h .

This suggests that the longer the time interval between the last two evaluations, the more significant impact the user’s latest evaluation has. Likewise, increased levels of familiarity, trust, and certainty contribute to a more substantial impact.

V. EXPERIMENTS

A. Experimental Setup

1) Environment: We conducted the experimental tests on two separate clusters: a GPU cluster for hosting the LLM part of the system and a CPU cluster dedicated to running the blockchain network. The first cluster comprises two servers, one featuring an NVIDIA RTX A6000 GPU card and the other equipped with an NVIDIA GeForce RTX 2080 Ti card. Meanwhile, the second cluster consists of two HPE ProLiant XL225n Gen10 Plus servers specifically allocated for experimenting with blockchain solutions. Each server in this cluster is powered by two AMD EPYC 7713 64-Core processors and 2x256 GB RAM.

2) Datasets: We evaluate LLMChain on three datasets:

- **MTBench**² is a recent dataset extensively utilized in evaluating LLMs [2]. MT-Bench consists of 3.3K expert-level pairwise human preferences for answers generated by six models (“Llama-13B”, “Alpaca-13B”, “Vicuna-13B”, “GPT-3.5”, “Claud-v1”, and “GPT-4”) across 80 questions.
- **GooAQ**³ is a large-scale dataset with a variety of answer types. This dataset comprises more than 5M questions and 3M answers sourced from Google [35].
- **LLMGooAQ**.⁴ We prepare this comprehensive database, covering 100k questions and answers in 20 different fields/context. We randomly sample 100K tuples from the GooAQ dataset and perform inference using seven LLMs (“Alpaca-13b”, “Llama-2-13b”, “Chatglm-6b”, “Fastchat-t5-3b”, “Koala-13b”, “Vicuna-7b”, “Vicuna-13b”).

3) Automatic Metrics: To pinpoint the optimal technique for our context, we conduct rigorous benchmarks among various embedding-based metrics that achieved SoTA performance.

- **BERTScore** [12] is an automatic evaluation metric for text generation. It evaluates the similarity between tokens in a candidate sentence and those in a reference sentence. Unlike N-Gram methods relying on exact matches like BLEU Score [36] and ROUGE Score [37], BERTscore relies on contextual embeddings to gauge token similarity. The approach employs cosine similarity to measure the likeness between a reference token x_i and a candidate token \hat{x}_i . The total score involves comparing each token in x with tokens in \hat{x} to calculate recall, and each token in \hat{x}_i with tokens in x to determine precision. To maximize the similarity score, a greedy matching technique is used, wherein each token is paired with the most similar token from the other sentence. Precision and recall are combined to derive an F1 score.
- **BARTScore** [30] is an automated evaluation method that frames the evaluation of generated text as a text generation

²<https://huggingface.co/spaces/lmsys/mt-bench>

³<https://huggingface.co/datasets/gooaq>

⁴<https://github.com/mohaminem/LLMGooAQ/>

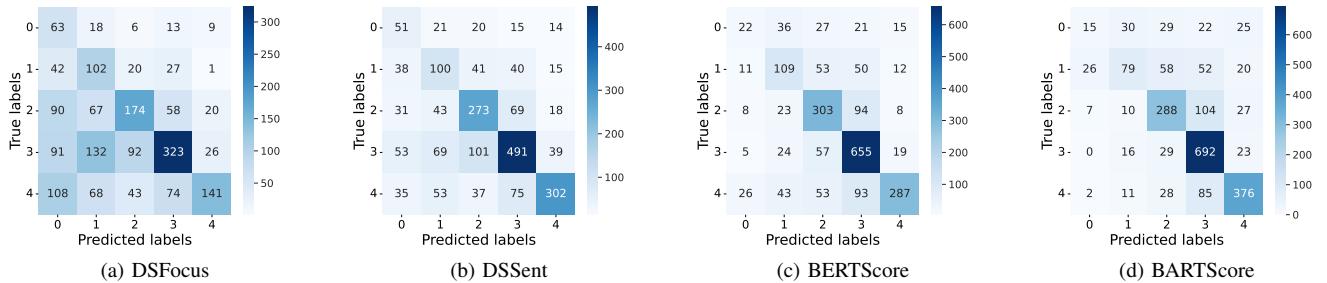


Fig. 3: -. Labels are denoted as: {“0:Llama-13B”, “1:Alpaca-13B”, “2:Vicuna-13B”, “3:GPT-3.5”, “4:Claud-v1”}

TABLE I: Hyperparameter’s Configuration.

Parameter	Value	Parameter	Value
ψ	1/3	$\alpha_r = \beta_r = \gamma_r$	1/3
ξ	2/3	$\alpha_u = \beta_u = \gamma_u$	1/3
λ	10^{-3}	θ	2/3

problem, utilizing pre-trained sequence-to-sequence models. The fundamental concept revolves around the notion that models trained to convert generated text into or from a reference output or the source text will yield higher scores for superior generated text. This concept is implemented using BART, a pre-trained model based on an encoder-decoder architecture. The metric BARTScore offers various adaptable variants that can be applied in an unsupervised manner to evaluate text from multiple perspectives, such as informativeness, fluency, or factuality.

- **DISCOScore** [31] is a parametrized discourse metric, which uses BERT to model discourse coherence from different perspectives, through the lens of readers’ focus, driven by Centering theory. DISCOScore offers two variations: FocusDiff and SentGraph, differing in their treatment of focus. This approach models the frequency and semantic relevance of focus and then compares the disparities between the hypothesis and the reference. It utilizes two adjacency matrices to represent coherence based on focus. In FocusDiff, the matrix represents relationships between foci and tokens, indicating focus frequency. Meanwhile, in SentGraph, the matrix showcases the interdependence between sentences based on shared foci and sentence proximity.

B. Reputation Model Effectiveness

In the following, we first perform an experimental comparison of the automatic metrics described in Sec. V-A3. Next, we perform two additional experiments aiming to evaluate the efficiency of both the automatic and human models. The values of the configurable parameters used in these experiments are summarized in Table. I.

- 1) **Metrics Benchmark.** Determining the most fitting metric for evaluating LLM-generated answers analytically is not straightforward. That is why we embarked on a benchmark experiment to pinpoint the best technique. This experiment

TABLE II: Metrics Performance on the MTBench dataset.

Metric	Accuracy	Kendall’s Correlation
DSFocus	0.44414	-0.60
DSSent	0.59540	0.60
BertScore	0.66991	0.60
BartScore	0.70594	0.80

aims to assess the metrics commonly used in automatically evaluating NLP tasks. Our goal is to identify the one that best aligns with human judgments. To achieve this, we conduct an experiment that involves computing automatic scores on MTBench answers. These scores automatically determine the winner between two different LLMs for each question. Fig. 3 demonstrates the correlation between human-selected winners (true) and automatic winners (predicted). The matrices show nearly diagonal patterns, indicating good correlations, yet variations in accuracy exist. For instance, the DISCOScore DSSent variant boasts an accuracy of 59%, surpassing that of the DSFocus variant (44%). BARTScore, on the other hand, demonstrates superior accuracy, with 71% of predicted winners matching actual human winners, compared with 67% for BERTScore. Table II illustrates Kendall’s Tau correlation of these four metrics. We can see that BARTScore can significantly outperform all other techniques by offering a superior correlation of 80% with human judgments. Based on these results, we decided to use **BARTScore** in the following experiments.

- 2) **Automatic Evaluation.** To adequately evaluate the automatic model, we use BARTScore to conduct a pairwise comparison between the seven LLMs in LLMGooAQ using GooAQ’s answers as benchmarks. Subsequently, we calculate the win rates for each LLM per context. The experimental results, showcased in Fig. 4, highlight “Vicuna-13b” as the best model outperforming others in nearly 90% of the contexts. Furthermore, the resulting models’ overall win rates align with previous human-based evaluation [2], affirming that the BARTScore metric correlates strongly with human judgments.

Now, to assess the efficacy of leveraging the best models’ answers within specific contexts, we conduct a subsequent test using the answers from “Vicuna-13b” as references.

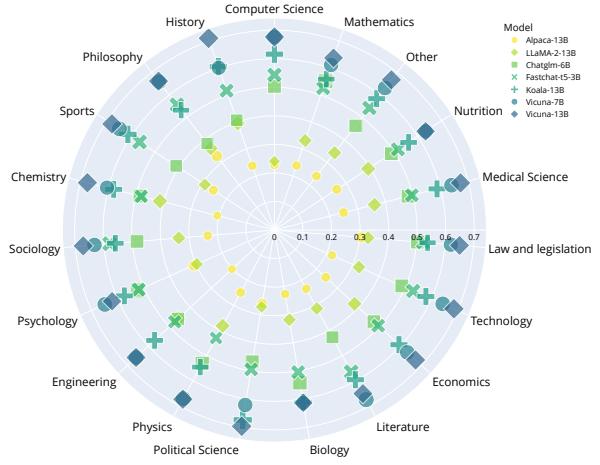


Fig. 4: BARTScore-based Contextual Win-Rates on LLMGooAQ.

Fig. 5 presents the confusion matrix comparing the winners (true) computed using GooAQ answers with those (predicted) computed using “Vicuna-13b” answers. The results are compelling, revealing robust accuracy (70%) between the two cases. It is essential to note that, according to current benchmarks [2], [6] and leaderboards (ChatBotArena⁵, TrustLLM⁶), “Vicuna-13b” is a well-ranked open source model, but it is not the best. Despite this, the results obtained using it as a reference model are convincing.

- 3) **Reputation Evaluation.** The third experiment involves employing the proposed models and monitoring changes in reputations in a real scenario. We use our prepared dataset with automatic scores computed using BARTScore to do this. Given the high cost of obtaining human judgments, we employ GPT-4 as an expert for human evaluation. GPT-4 is recognized as the leading model in current benchmarks [2], [6], [21]. In this experiment, GPT-4 is used to play the role of a human expert, responding to a questionnaire that enables the calculation of metrics (*i.e.* F , T , U , A_t , A_c , and A_u) used in the human model. Fig. 6 illustrates the variations in R^a , R^h , and R for the seven LLMs in our dataset. Despite the disparities between the R^a and R^h scores, a consistent pattern emerges, with scores for good models such as “Koala-13b”, “Vicuna-7b”, and “Vicuna-13b” steadily increasing, while scores for less effective models such as “Alpaca-13b” and “Llama-2-13b” continually decrease. Moreover, with an increasing number of evaluations, the distinctions between closely ranked models become more pronounced. This demonstrates the effectiveness of our models, showcasing their ability to discern even subtle differences between close LLMs like “Chatglm-6b” and “Fastchat-t5-3b”.

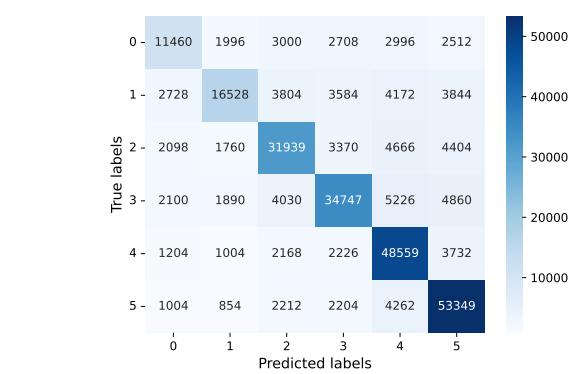


Fig. 5: Ground-Truth Answers vs Vicuna-13B Answers as References for BARTScore-based Pairwise-comparison on the LLM-GooAQ dataset. Labels are denoted as: {0: “Alpaca-13b”, 1: “Llama-2-13b”, 2: “Chatglm-6b”, 3: “Fastchat-t5-3b”, 4: “Koala-13b”, 5: “Vicuna-7b”, 6: “Vicuna-13b”}.

C. Blockchain Performance

1) **Business Model:** Having evaluated all its components in the previous subsection, we now implement the proposed blockchain-driven framework. This one is deployed on a blockchain network powered by Hyperledger Besu⁷, an open-source Ethereum client. Our evaluation approach includes:

- Participants: Users with different expertise and Admins of the organization or the consortium operating the system.
- Assets: A data structure that represents the model on-chain.
- Smart Contracts: Three types of smart contracts are used to develop the business model: Identity Smart Contract (ISC), Access Control Smart Contract (ACSC), and Reputation Smart Contract (RSC). ISC implements the registration process, ACSC employs a role-based access control to manage the permissions when calling RSC functions, *e.g.* only Oracles can trigger the *autoEval* function. The RSC implements four main functions, *addModel*, *autoEval*, *humEval*, and *updateReputation*.

We develop the smart contracts of LLMChain using the Solidity programming language⁸ and establish a local network consisting of sixteen validators using Hyperledger Besu with Proof of Authority (PoA) as consensus protocol. We lastly use Web3js library⁹ for developing the client side and deploying the system’s smart contracts.

2) **Performance Evaluation:** To conduct tests, we utilize Hyperledger Caliper¹⁰, a benchmarking tool for blockchains. The experiments involve changing the transaction sending rate (ranging from 50 to 1000 TPS) using a consistent network configuration for the main operations performed within LLM-Chain. As a result, two metrics are measured:

- **Throughput:** is the number of successful transactions per second (TPS).

⁵<https://huggingface.co/spaces/lmsys/chatbot-area-leaderboard>

⁶<https://trustllmbenchmark.github.io/TrustLLM-Website/leaderboard.html>

⁷<https://besu.hyperledger.org>

⁸<https://docs.soliditylang.org>

⁹<https://web3js.readthedocs.io>

¹⁰<https://github.com/hyperledger/caliper-benchmarks>

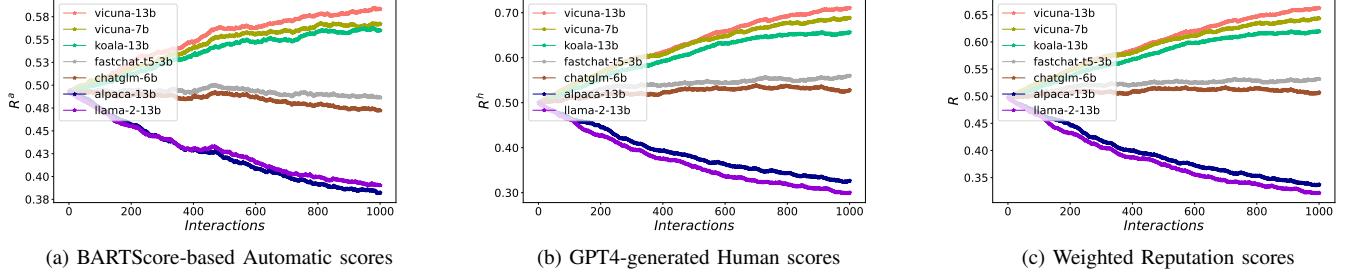


Fig. 6: Changes in R^a , R^h , and R of seven LLMs using LLMGooAQ.

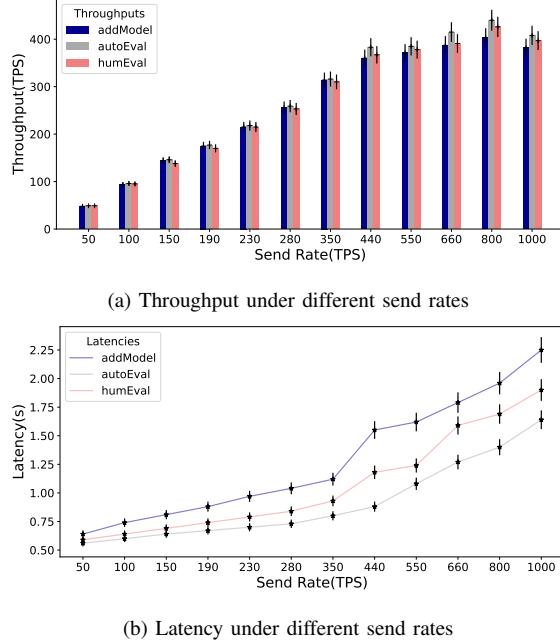


Fig. 7: Throughput and Latency of LLMChain.

- **Latency:** refers to the time difference in seconds between the submission and completion of a transaction.

The throughput and latency values for each function under different sending rates are illustrated in Fig. 7. At the beginning, the pattern is evident: throughput and latency increase as the transaction send rate increases. With lower sending rates (< 350 TPS), there is no significant difference in throughput between the three defined transaction types. However, nearing system capacity, distinctions emerge. The lightest function, *autoEval*, achieves a peak throughput of 440 TPS, surpassing *humEval* at 426 TPS, and the heaviest function, *addModel*, managing 403 TPS, primarily due to the initialization and storage of model information on-chain. This also explains the comparatively higher latency of *addModel* compared with the other two functions. Nevertheless, leveraging storage scaling via IPFS, LLMChain achieves an average throughput close to 420 TPS, comfortably meeting the specific demands of our use case. On top of that, since LLMChain operates on an EVM-based state machine, all the scaling techniques of

Ethereum-based blockchains, such as Sharding and zkRollups can be applied to further enhance its performance for large-scale deployment if needed.

VI. LIMITATIONS AND FUTURE RESEARCH DIRECTIONS

To the best of our knowledge, we are the first to design and develop a reputation model for evaluating LLMs within a decentralized framework. While our experiments prove the effectiveness and scalability of LLMChain, we believe that this work promotes future research on decentralized and transparent language model evaluation. However, LLMChain presents some limitations regarding both human and automatic evaluations. Firstly, human evaluation depends mainly on users' willingness to provide authentic feedback. Further assurance and incentive measures can be added to the framework to improve the reliability of human evaluation. Secondly, automatic evaluation relies on the availability of reference models. This approach has proved effective. However, it has two important shortcomings: i) Its accuracy depends on the performance of available reference models, ii) and even if the k responses can help the user to provide a better human evaluation, this approach generates off-chain communication and computational overheads.

VII. CONCLUSION

In this paper, we propose LLMChain, a novel blockchain-powered framework, specifically designed to share and evaluate LLMs efficiently and transparently. LLMChain addresses trust concerns associated with flawed behaviors like hallucinations and unreliable reasoning of LLMs by employing a context-driven reputation system. Our efforts involve crafting and implementing a reputation model that evaluates user satisfaction and trust in each interaction involving an LLM. This model amalgamates human feedback with automatic evaluation to assign contextual reputation scores that accurately mirror LLM behavior. Consequently, the system aids users and entities in pinpointing the most credible LLM for their requirements while offering LLM providers valuable insights to refine and enhance their models. This research marks the first initiative to introduce a distributed framework dedicated to LLMs evaluation. Through extensive experiments and benchmarks, we demonstrate the effectiveness of both human and automatic evaluations in LLMChain. Moreover, the tests

conducted on the deployed blockchain affirm LLMChain's efficiency and scalability, validating its practical applicability in real-world scenarios. Finally, LLMGooAQ, a large dataset of over 100K questions and answers generated using seven LLMs, was prepared and released to the community to advance research in this area further.

ACKNOWLEDGEMENT

This work was supported by the 5G-INSIGHT bilateral ANR-FNR project , the Nouvelle-Aquitaine Region - B4IoT project, the French government in the framework of the France Relance program, and the ITSOFT company under grant number AD 22-252.

REFERENCES

- [1] C. Qin, A. Zhang, Z. Zhang, J. Chen, M. Yasunaga, and D. Yang, "Is ChatGPT a general-purpose natural language processing task solver?" in *2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 1339–1384.
- [2] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li *et al.*, "Judging LLM-as-a-judge with MT-bench and chatbot arena," in *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.
- [3] R. Dale, "Gpt-3: What's it good for?" *Natural Language Engineering*, vol. 27, no. 1, pp. 113–118, 2021.
- [4] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.
- [5] W.-L. Chiang, Z. Li, Z. Lin, Y. Sheng, Z. Wu, H. Zhang *et al.*, "Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality," March 2023.
- [6] L. Sun, Y. Huang, H. Wang, S. Wu, Q. Zhang, C. Gao, Y. Huang, W. Lyu, Y. Zhang, X. Li *et al.*, "Trustilm: Trustworthiness in large language models," *arXiv preprint arXiv:2401.05561*, 2024.
- [7] C. Wang, X. Liu, Y. Yue, X. Tang, T. Zhang, C. Jiayang, Y. Yao, W. Gao, X. Hu, Z. Qi *et al.*, "Survey on factuality in large language models: Knowledge, retrieval and domain-specificity," *arXiv preprint arXiv:2310.07521*, 2023.
- [8] J. Li, X. Cheng, W. X. Zhao, J.-Y. Nie, and J.-R. Wen, "Halueval: A large-scale hallucination evaluation benchmark for large language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023, pp. 6449–6464.
- [9] O. Golovneva, M. Chen, S. Poff, M. Corredor, L. Zettlemoyer *et al.*, "ROSCOE: A suite of metrics for scoring step-by-step reasoning," in *The Eleventh International Conference on Learning Representations, ICLR Kigali, Rwanda*. OpenReview.net, 2023.
- [10] L. Pan, M. Saxon, W. Xu, D. Nathani, X. Wang, and W. Y. Wang, "Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies," 2023.
- [11] P. Fernandes, A. Madaan, E. Liu, A. Farinhas, P. H. Martins, A. Bertsch *et al.*, "Bridging the Gap: A Survey on Integrating (Human) Feedback for Natural Language Generation," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 1643–1668, 12 2023.
- [12] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with BERT," in *8th International Conference on Learning Representations, ICLR, Addis Ababa, Ethiopia*, 2020.
- [13] J. Belouadi and S. Eger, "UScore: An effective approach to fully unsupervised evaluation metrics for machine translation," in *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, A. Vlachos and I. Augenstein, Eds. Dubrovnik, Croatia: Association for Computational Linguistics, May 2023, pp. 358–374.
- [14] H. Schuff, L. Vanderlyn, H. Adel, and N. T. Vu, "How to do human evaluation: A brief introduction to user studies in nlp," *Natural Language Engineering*, vol. 29, no. 5, p. 1199–1222, 2023.
- [15] C.-H. Chiang and H. yi Lee, "Can large language models be an alternative to human evaluations?" 2023.
- [16] Y. Liu, D. Iter, Y. Xu, S. Wang, R. Xu, and C. Zhu, "G-eval: NLG evaluation using gpt-4 with better human alignment," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 2511–2522.
- [17] S. Malik, V. Dedeoglu, S. S. Kanhere, and R. Jurdak, "Trustchain: Trust management in blockchain and iot supported supply chains," *IEEE International Conference on Blockchain*, pp. 184–193, 2019.
- [18] M. Li, J. Weng, A. Yang, W. Lu, Y. Zhang, L. Hou, J.-N. Liu, Y. Xiang, and R. H. Deng, "Crowdbc: A blockchain-based decentralized framework for crowdsourcing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 6, pp. 1251–1266, 2019.
- [19] M. A. Bouchiba, Y. Ghamri-Doudane, M. Rabah, and R. Champagnat, "Guruchain: Guarantee and reputation-based blockchain service trading platform," in *IFIP Networking Conference*, 2023, pp. 1–9.
- [20] A. Köpf, Y. Kilcher, D. von Rütte, S. Anagnostidis, Z.-R. Tam, K. Stevens, A. Barhoum *et al.*, "Openassistant conversations – democratizing large language model alignment," 2023.
- [21] D. Pride, M. Cancellieri, and P. Knoth, "Core-gpt: Combining open access research and large language models for credible, trustworthy question answering," in *International Conference on Theory and Practice of Digital Libraries*. Springer, 2023, pp. 146–159.
- [22] W. Ye, M. Ou, T. Li, X. Ma, Y. Yanggong, S. Wu, J. Fu, G. Chen, J. Zhao *et al.*, "Assessing hidden risks of llms: An empirical study on robustness, consistency, and credibility," *arXiv preprint arXiv:2305.10235*, 2023.
- [23] R. Lai and G. Zhao, "Validatorrep: Blockchain-based trust management for ensuring accountability in crowdsourcing," in *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE, 2022, pp. 716–725.
- [24] J. Arshad, M. A. Azad, A. Prince, J. Ali, and T. G. Papaioannou, "Reputable—a decentralized reputation system for blockchain-based ecosystems," *IEEE Access*, vol. 10, pp. 79 948–79 961, 2022.
- [25] Z. Jaroucheh, M. Alissa, W. J. Buchanan, and X. Liu, "Trustd: Combat fake content using blockchain and collective signature technologies," in *2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE, 2020, pp. 1235–1240.
- [26] X. Zhu, Y. Li, L. Fang, and P. Chen, "An improved proof-of-trust consensus algorithm for credible crowdsourcing blockchain services," *IEEE Access*, vol. 8, pp. 102 177–102 187, 2020.
- [27] J. Benet, "Ipfs - content addressed, versioned, p2p file system," 2014.
- [28] L. Breidenbach, C. Cachin, B. Chan, A. Coventry, S. Ellis, A. Juels, F. Koushanfar, A. Miller, B. Magauran, D. Moroz *et al.*, "Chainlink 2.0: Next steps in the evolution of decentralized oracle networks," *Chainlink Labs*, vol. 1, pp. 1–136, 2021.
- [29] D. Maram, H. Malvai, F. Zhang, N. Jean-Louis, A. Frolov, T. Kell *et al.*, "Candid: Can-do decentralized identity with legacy compatibility, sybil-resistance, and accountability," in *2021 IEEE Symposium on Security and Privacy (SP)*, 2021, pp. 1348–1366.
- [30] W. Yuan, G. Neubig, and P. Liu, "Bartscore: Evaluating generated text as text generation," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Curran Associates, Inc., 2021, pp. 27 263–27 277.
- [31] W. Zhao, M. Strube, and S. Eger, "DiscoScore: Evaluating text generation with BERT and discourse coherence," in *in 17th Conference of the European Chapter of the Association for Computational Linguistics*. Dubrovnik, Croatia: Association for Computational Linguistics, May 2023, pp. 3865–3883.
- [32] M. Körber, "Theoretical considerations and development of a questionnaire to measure trust in automation," in *Proceedings of the 20th Congress of the International Ergonomics Association (IEA 2018) Volume VI: Transport Ergonomics and Human Factors (TEHF), Aerospace Human Factors and Ergonomics 20*. Springer, 2019, pp. 13–30.
- [33] Y. Zhang and M. van der Schaar, "Reputation-based incentive protocols in crowdsourcing applications," in *2012 Proceedings IEEE INFOCOM*, 2012, pp. 2140–2148.
- [34] E. Bellini, Y. Iraqi, and E. Damiani, "Blockchain-based distributed trust and reputation management systems: A survey," *IEEE Access*, vol. 8, pp. 21 127–21 151, 2020.
- [35] D. Khashabi, A. Ng, T. Khot, A. Sabharwal, H. Hajishirzi, and C. Callison-Burch, "Gooaq: Open question answering with diverse answer types," *arXiv preprint arXiv:2104.08727*, 2021.
- [36] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, Jul. 2002, p. 311–318.
- [37] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81.

EEG-Deformer: A Dense Convolutional Transformer for Brain-computer Interfaces

Yi Ding, *Member, IEEE*, Yong Li, Hao Sun, Rui Liu, Chengxuan Tong, *Graduate Student Member, IEEE*, and Cuntai Guan, *Fellow, IEEE*

Abstract—Effectively learning the temporal dynamics in electroencephalogram (EEG) signals is challenging yet essential for decoding brain activities using brain-computer interfaces (BCIs). Although Transformers are popular for their long-term sequential learning ability in the BCI field, most methods combining Transformers with convolutional neural networks (CNNs) fail to capture the coarse-to-fine temporal dynamics of EEG signals. To overcome this limitation, we introduce EEG-Deformer, which incorporates two main novel components into a CNN-Transformer: (1) a Hierarchical Coarse-to-Fine Transformer (HCT) block that integrates a Fine-grained Temporal Learning (FTL) branch into Transformers, effectively discerning coarse-to-fine temporal patterns; and (2) a Dense Information Purification (DIP) module, which utilizes multi-level, purified temporal information to enhance decoding accuracy. Comprehensive experiments on three representative cognitive tasks consistently verify the generalizability of our proposed EEG-Deformer, demonstrating that it either outperforms existing state-of-the-art methods or is comparable to them. Visualization results show that EEG-Deformer learns from neurophysiologically meaningful brain regions for the corresponding cognitive tasks. The source code can be found at <https://github.com/yi-ding-cs/EEG-Deformer>.

Index Terms—Deep learning, electroencephalography, transformer.

I. INTRODUCTION

BRAIN-computer interface (BCI) technology facilitates direct communication between the brain and machines using electroencephalography (EEG) [1]. A standard BCI system usually consists of four key components: data acquisition, pre-processing, classification, and feedback [2]. BCIs are employed in various practical applications, such as stroke rehabilitation [3] and emotion regulation in mental health treatments [4].

EEG signals, collected through electrodes placed on each subject's head, comprise spatial and temporal dimensions. The spatial dimension relates to the locations of the EEG electrodes, while the temporal dimension captures fluctuations in brain activity [2]. To ensure accuracy in decoding neural activity, a reliable BCI system must effectively perceive the subtle temporal dynamics encoded in EEG signals, which implicitly represent various cognitive processes. However, this

Yi Ding and Yong Li contribute equally to this work.

Yi Ding, Yong Li, Rui Liu, Chengxuan Tong, and Cuntai Guan are with the School of Computer Science and Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore, 639798.

Hao Sun is with the Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, East China University of Science and Technology, Shanghai, China.

Chengxuan Tong is with Wilmar International, Singapore.

Cuntai Guan is the Corresponding Author.

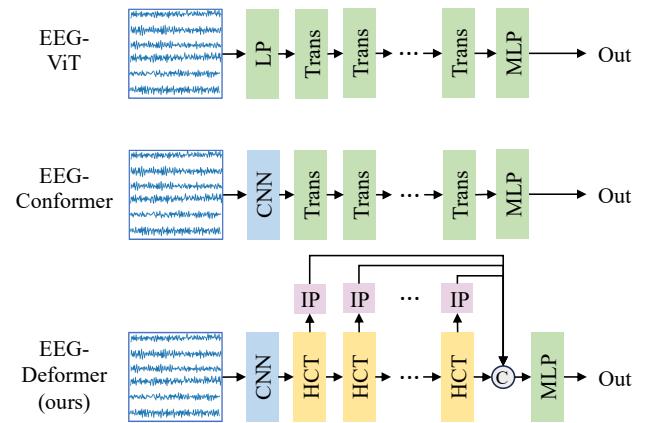


Fig. 1. Comparison of network architectures between ViT [5], EEG-Conformer [6], and our proposed **EEG-Deformer**. We propose a novel Hierarchical Coarse-to-Fine Transformer (HCT). Additionally, we have designed Information Purification Unit (IP-Unit, denoted by IP in the figure) for each HCT layer with dense connections to further boost EEG decoding performance.

task is quite challenging, as these brain activities vary across subjects and are susceptible to external interference, such as movement artifacts, eye blinks, and environmental factors [7].

Recently, numerous deep learning-based approaches have been developed for decoding brain activities from EEG signals. These approaches can be broadly categorized into two types: methods using hand-crafted features and methods using EEG directly as input. The former [8], [9], [10], [11] involve extracting various types of features from EEG signals for neural network input. Meanwhile, CNN-based methods, which leverage automatic feature learning, typically process EEG as 2-D time series [12], [13], [14]. Given that EEG data is inherently graph-structured, with nodes representing electrodes and connections based on spatial distance, functional connectivity, or learned relationships, graph-based methods have gained popularity [8], [15]. However, using features as input may result in the loss of fine-grained temporal information by averaging data along the temporal dimensions. Additionally, CNN-based methods may fail to capture long-term temporal dependencies by employing CNN kernel along temporal dimension.

In addition to CNNs, transformer-based neural architectures have attracted significant attention in the BCI field due to their inherent ability to perceive global dependencies. Commonly, prior works [16], [17], [18], [6] adopt a CNN-Transformer architecture, where the CNN part serves as an adaptive feature

encoder to preprocess EEG data, and the subsequent transformer part captures long-range temporal characteristics. The CNN-based shallow feature encoder has been verified as essential for preparing EEG data for Transformers [16]. Although previous works can capture either fine-grained (short-period) or coarse-grained (global) temporal dependencies within each layer, they have not explicitly captured both coarse- and fine-grained temporal dynamics within the Transformer layers, which may limit the full utilization of EEG signals' long-short period temporal dynamics [19]. Moreover, most existing methods overlook the abundant latent features encoded within intermediate neural layers. These layers encapsulate rich temporal information that can be systematically explored to enhance the precise discernment of temporal dynamics inherent in EEG data.

To mitigate the above-mentioned issues and enhance the perception of temporal dynamics in EEG data, we introduce EEG-Deformer, a novel dense convolutional Transformer. We propose a novel Hierarchical Coarse-to-Fine Transformer (HCT) block that integrates a Fine-grained Temporal Learning (FTL) branch into Transformers. Built upon a CNN-based shallow feature encoder comprising collaborative temporal and spatial convolutional layers, HCT concurrently captures coarse- and fine-grained temporal dynamics, as shown in Figure 1. The FTL branch employs a 1-D CNN to sequentially capture short-period temporal dynamics of EEG, generating fine-grained temporal representations. These representations are then adaptively fused with the coarse-grained temporal representations encoded by the Transformers, thereby providing more discriminative long- and short-term temporal information.

Furthermore, to efficiently utilize multi-level temporal information from intermediate HCT layers, we have designed a Dense Information Purification (DIP) module in our EEG-Deformer. This module enables the dense transmission of multi-level representations from HCT layers to the final representation. Differing from the skip connections in DenseNet [20], we introduce an Information Purification Unit (IP-Unit) that transforms the fine-grained representations from FTL branches using a logarithmic (log) power operation [15] and elegantly fuses these latent representations into the final representation. Because log power can represent the amount of activity in filtered signals and reduce dimensions [21], our IP-Unit not only retains critical frequency characteristics related to brain activity but also effectively reduces the number of learnable parameters. As illustrated in Figure 1, a series of IP-Units have been added to the intermediate layers. These units progressively encode discriminative temporal representations. We will validate these enhancements in Sec. V-B to V-E.

In summary, the contributions of this work are summarized as follows:

- We introduce EEG-Deformer, a novel architecture designed for EEG decoding across various cognitive tasks.
- A hierarchical coarse-to-fine Transformer is proposed to effectively encode the coarse-to-fine temporal dynamics within EEG data.
- We develop a dense information purification module to fully exploit abundant intermediate multi-level EEG

features and enhance the EEG decoding performance.

- Through extensive experimentation on three public datasets, encompassing attention, fatigue, and cognitive workload classification tasks, the efficacy of EEG-Deformer is demonstrated. Our results show its superiority compared to current state-of-the-art (SOTA) methods.

II. RELATED WORK

A. CNNs for EEG Decoding

CNNs have emerged as potent tools in BCI applications, adept at directly learning from EEG data. Schirrmeister *et al.* [12] introduced DeepConvNet, which employs a dual-stage spatial and temporal convolution layer for EEG data feature extraction and classification. In parallel, Lawhern *et al.* [13] created EEGNet, utilizing depth-wise convolution with a kernel size of $(n, 1)$, where n signifies the number of EEG channels, to capture spatial information. Advancing this concept, TSception [14] leverages multi-scale convolutional kernels to decode EEG's temporal dynamics and asymmetric spatial activations. However, the relatively short length of these 1-D CNN kernels in the temporal dimension slightly curtails their efficacy in learning long-term temporal patterns.

B. Transformers for BCI

Transformers, known for their ability to learn long-term correlations in sequential data, have garnered increasing attention from researchers [16], [17], [18], [6]. Lee *et al.* [18] demonstrate that incorporating a self-attention module from Transformer architecture into an EEGNet-based CNN enhances the classification of imagined speech using EEG. EEG-Conformer [6], a compact convolutional Transformer, effectively merges local and global features for EEG decoding in a cohesive framework. However, most existing convolutional Transformers, which typically use CNNs for shallow feature extraction followed by Transformer blocks, may not efficiently learn the coarse- and fine-grained temporal information encoded in EEG signals. Additionally, they often overlook the multi-level temporal information available from different layers.

III. METHOD

In this work, we propose a novel Transformer, EEG-Deformer, for general EEG decoding in the BCI field. The network architecture is shown in Figure 2. EEG-Deformer consists of three main components: (1) a Shallow feature encoder, (2) a Hierarchical Coarse-to-fine Transformer (HCT), and (3) a Dense Information Purification (DIP) module. Given an input segment of EEG signals, EEG-Deformer utilizes the CNN feature encoder to adaptively encode the shallow temporal and spatial features, which are then set as input into the following HCT blocks to extract the temporal dynamics that happen in different timescales in EEG signals. To effectively perceive the critical multi-level temporal information, the features generated from each HCT block are adaptively fused via progressive IP-Units. Below, we present the details for each of them.

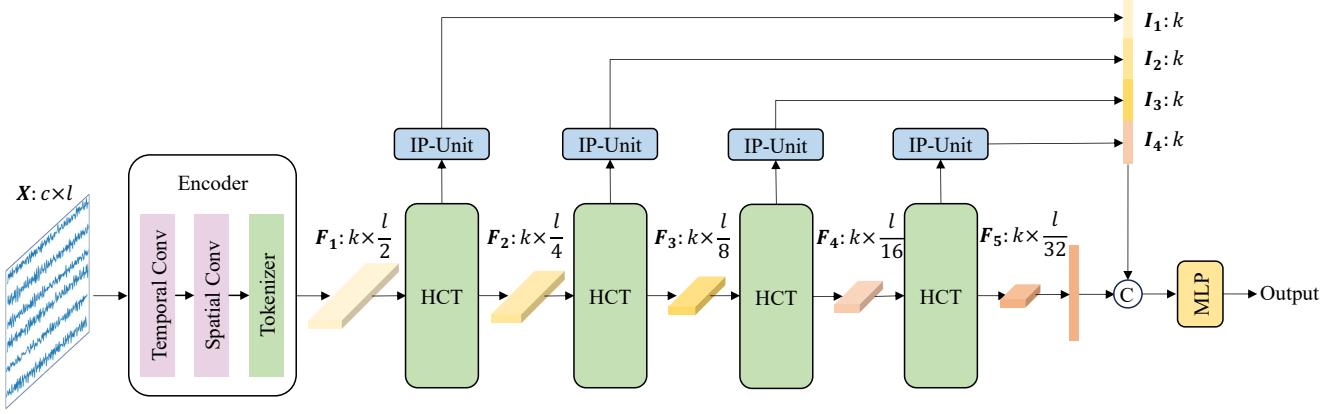


Fig. 2. The network structure of EEG-Deformer. EEG-Deformer consists of three main parts: (1) Shallow feature encoder, (2) Hierarchical coarse-to-fine-Transformer (HCT), and (3) Dense information purification (DIP). The fine-grained representations from each HCT will be passed to Information Purification Unit (IP-Unit) and concatenated to the final embedding.

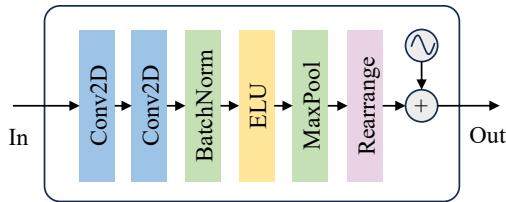


Fig. 3. The structure of the shallow feature encoder. After standard CNN layers, the representation is rearranged into kernel by feature, and a position encoding is added onto it.

A. Shallow Feature Encoders

To capture the shallow temporal and spatial information of EEG, we utilize a CNN-based shallow feature encoder. It comprises two main components: CNN layers and a tokenizer. CNNs along temporal and spatial dimensions are commonly used as feature extractors for EEG signals [13]. In our proposed EEG-Deformer, a two-layer CNN is adopted as the shallow feature encoder. The architecture of the CNN encoder is shown in Figure 3. It begins with temporal and spatial CNN layers, followed by batch normalization to mitigate the covariate shift issue.

Let's denote an EEG sample as $\mathbf{X} \in \mathbb{R}^{c \times l}$, where c is the number of EEG channels, and l represents the number of data points along the time dimension. Inspired by neurophysiological knowledge that the brain has microstates lasting approximately 100 ms, the temporal CNN kernels are sized at $(1, 0.1 \times f_s)$, where f_s denotes the EEG's sampling rate. For capturing spatial information from EEG, a spatial CNN kernel of size $(c, 1)$ is utilized as suggested in [13], where c is the number of EEG channels. Weight normalization is included following the approach in [22]. The number of CNN kernels is denoted by k . After activation by the ELU function, the learned features are max-pooled every two data points without overlapping. The tokenizer includes rearrange operation and a learnable position encoding. The size of the features is then rearranged into $k \times 0.5l$ to serve as tokens for the Trans-

formers, which learn coarse-grained temporal information. Subsequently, a learnable position encoding, $\mathbf{P} \in \mathbb{R}^{k \times 0.5l}$, is added to the tokens [5]. Therefore, the encoded tokens can be represented as $\mathbf{F} \in \mathbb{R}^{k \times 0.5l}$.

$$\mathbf{F} = \Gamma(\text{MaxPool}(\text{ELU}(\text{BN}(\text{CNN}(\mathbf{X})))) + \mathbf{P}, \quad (1)$$

where $\Gamma(\cdot)$ is the rearrange operation and $\text{BN}(\cdot)$ is the batch normalization layer.

B. Hierarchical Coarse-to-fine Transformer

With the encoded shallow EEG features, we aim to learn the coarse and fine-grained temporal dynamics in EEG data via cascading HCT blocks. The neural structure of a HCT block is shown in Fig. 4. A HCT consists parallel Transformer-based branch that aims to learn the correlations among the given tokens [23] and CNN-based branch that aims to learn fine-grained EEG features.

Let us suppose $\mathbf{F}_i \in \mathbb{R}^{k \times l_i^i}$ the input to the i -th HCT block. To capture the coarse-grained temporal dynamics of EEG signals, we treat the output of each CNN kernel as one token. By doing so, the long-term temporal information is explicitly included when we project the tokens into $\mathbf{Q}_i \in \mathbb{R}^{n_{\text{head}} \times k \times d_{\text{attn}}}$, $\mathbf{K}_i \in \mathbb{R}^{n_{\text{head}} \times k \times d_{\text{attn}}}$, and $\mathbf{V}_i \in \mathbb{R}^{n_{\text{head}} \times k \times d_{\text{attn}}}$ using the linear projection (LP) layer parameterized by n_{head} set of $\mathbf{W}_{qkv}^i \in \mathbb{R}^{2^{i-1} \times 3d_{\text{attn}}}$. Before the LP layer, a max pooling layer is added to reduce the feature dimension of \mathbf{F}_i . The encoding process $\Phi(\cdot)$ can be formulized by

$$\{\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i\} = \Phi_i(\mathbf{F}_i) = \text{MaxPool}(\mathbf{F}_i)\mathbf{W}_{qkv}^i. \quad (2)$$

A multi-head self-attention (MSA) is then utilized to extract the correlations among the different views of the coarse-grained temporal embeddings. The scaled dot-product is utilized as the attention operation along temporal tokens for each attention head.

$$\text{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}(\mathbf{Q}\mathbf{K}^T / \sqrt{d})\mathbf{V}, \quad (3)$$

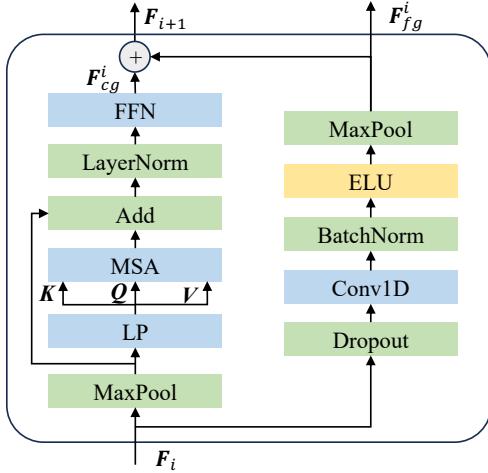


Fig. 4. The structure of the hierarchical coarse-to-fine Transformer: The right side is capable of learning coarse-grained temporal information through self-attention, while the left side is the newly added FTL.

where $d = d_{attn}$ is a scaling factor. The output of each attention head will be concatenated and linearly projected into a tensor, \mathbf{F}_{msa}^i , that has the same size of the input by

$$\begin{aligned}\mathbf{F}_{msa}^i &= \text{MSA}(\mathbf{F}_i) \\ &= [\text{Attn}(\Phi_i^1(\mathbf{F}_i)), \dots, \text{Attn}(\Phi_i^{n_{head}}(\mathbf{F}_i))] \mathbf{W}_{attn}^i,\end{aligned}\quad (4)$$

where $[\cdot]$ is concatenation operation and $\mathbf{W}_{attn}^i \in \mathbb{R}^{n_{head}d_{attn} \times \frac{l_t^i}{2^{i-1}}}$ is the learnable weights.

After the MSA, a residual connection is introduced, followed by layer normalization and a Feedforward Neural Network (FFN). The FFN consists of two layers, utilizing a GELU non-linearity, similar to those used in ViT [5]. Consequently, the coarse-grained embedding, denoted as \mathbf{F}_{cg}^i , is generated by:

$$\mathbf{F}_{cg}^i = \text{FFN}(\text{LN}(\mathbf{F}_{msa}^i + \text{MaxPool}(\mathbf{F}_i))), \quad (5)$$

where $\text{LN}(\cdot)$ is the layer normalization operation.

We further propose a FTL module to capture the short-period temporal dynamics of EEG signals. To learn fine-grained temporal features, we opt for a 1D CNN, as its short kernel moves along the temporal dimension step by step, enabling the learning of short-period patterns. After a dropout layer, the learned representations are fed into a 1-D CNN layer, followed by a batch normalization layer, an ELU activation, and a max-pooling layer. The fine-grained temporal representations, denoted as \mathbf{F}_{fg}^i , can be calculated as follows:

$$\mathbf{F}_{fg}^i = \text{MaxPool}(\text{ELU}(\text{BN}(\text{CNN}(\text{DP}(\mathbf{F}_i)))), \quad (6)$$

where $\text{DP}(\cdot)$ is the dropout operation.

After learning the coarse- and fine-grained temporal representations, a sum fusion is added to get the final output of the HCT layer

$$\mathbf{F}_{i+1} = \mathbf{F}_{cg}^i + \mathbf{F}_{fg}^i. \quad (7)$$

For the \mathbf{F}_{fg}^i , it is also used to do the information purification process.

C. Dense Information Purification

With the learned fine-grained temporal features from different HCT layers, we further utilize densely connected IP-Units to progressively extract the multi-level temporal information from these layers. It aims to extract discriminative information from a frequency perspective and reduce the size of the bypassed multi-level representations.

Drawing inspiration from neural engineering, where power features of EEG signals in different frequency bands are widely used for brain activity analysis [24], [12], we propose a power layer for information purification and encode frequency information in EEG signals. The power of the learned 1-D hidden representations, $\mathbf{I}_i \in \mathbb{R}^k$, can be calculated by

$$\begin{aligned}\mathbf{I}_i &= \text{IP}_{power}(\mathbf{F}_{fg}^i) \\ &= \{\log\left(\frac{1}{l_t^i} \sum (\mathbf{f}_{fg}^{i,j})^2\right) : \mathbf{f}_{fg}^{i,j} \in \mathbf{F}_{fg}^i\},\end{aligned}\quad (8)$$

where $\log(\cdot)$ is the logarithm, and $\mathbf{f}_{fg}^{i,j} \in \mathbb{R}^{l_t^i}$ is one learned representation from the j -th CNN kernel in the FTL.

The purified information from all HCT layers is concatenated with the flattened output of the last HCT layer to obtain the final hidden embedding. A linear layer is then utilized as the classifier to project the hidden embedding onto the class labels. Let n_{hct} denote the total number of HCT layers; this process can be formulated as:

$$\mathbf{out} = [\mathbf{F}_{n_{hct}+1}, \mathbf{I}_1, \dots, \mathbf{I}_{n_{hct}}] \mathbf{W} + \mathbf{b}, \quad (9)$$

where $[\cdot]$ is the concatenation operation, \mathbf{W} and \mathbf{b} are the trainable weights and biases.

IV. EXPERIMENT

A. Datasets

We evaluate the performance of EEG-Deformer with three benchmarking EEG datasets, which are Dataset I [25] for cognitive attention, Dataset II [26] for driving fatigue, and Dataset III [27] for cognitive workload.

Dataset I: Dataset I provides EEG signals recorded while subjects perform the Discrimination/Selection Response (DSR) task to assess cognitive attention. The experiment involves a total of 26 participants. Each subject undergoes three sessions, with each session consisting of multiple cycles of attention tasks, each lasting 40 seconds, and alternating with 20-second rest periods. Data collection includes recordings from 28 EEG channels, sampled at a rate of 1000 Hz. According to [15], the first 20 seconds of attention tasks are used to balance the classes. All three sessions are utilized in this study. Each EEG trial is segmented using a sliding window of 4 seconds with a 50% overlap.

Dataset II: Dataset II is designed to evaluate cognitive fatigue states in drivers through EEG signal analysis during a 90-minute driving task within a virtual reality (VR) driving environment. This dataset was compiled with the participation of 27 subjects. EEG data were captured using a 32-channel

TABLE I

GENERALIZED CROSS-SUBJECT CLASSIFICATION RESULTS FROM VARIOUS METHODS ON THREE BENCHMARKING DATASETS. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD AND THE NEXT BEST ARE MARKED USING UNDERLINES

Method	Dataset I				Dataset II				Dataset III			
	ACC	std	F1-macro	std	ACC	std	F1-macro	std	ACC	std	F1-macro	std
DGCNN	60.98	6.05	56.81	9.51	69.56	12.50	64.92	13.08	64.85	17.90	59.42	22.25
LGGNet	67.81	6.38	66.70	7.69	68.16	13.16	65.38	12.40	65.37	12.78	61.91	16.16
EEGNet	75.28	9.77	73.81	12.62	73.95	12.54	71.43	13.45	65.85	14.55	61.89	18.21
TSception	71.60	7.66	69.94	10.88	73.01	12.93	70.18	13.90	69.92	15.07	65.26	20.04
EEG-ViT	67.72	7.03	66.79	8.49	66.95	14.12	64.12	13.89	63.12	15.44	61.19	16.56
SSVEPformer	73.18	7.87	72.7	8.54	55.02	10.95	52.00	9.87	55.17	6.71	54.25	7.01
EEG-Transformer	75.39	7.92	74.93	8.61	65.53	11.2	63.86	12.84	64.46	14.91	64.28	14.94
EEG-Conformer	79.81	9.27	79.01	10.75	74.36	11.82	71.65	12.96	69.40	14.48	65.59	18.68
EEG-Deformer (ours)	82.72	8.00	82.36	8.52	79.32	7.87	75.83	11.54	73.18	15.63	69.99	19.80

TABLE II

RESULTS OF THE ABLATION STUDY CONDUCTED ON THREE BENCHMARKING DATASETS, WITH THE BEST RESULTS MARKED IN BOLD.

Method	Dataset I				Dataset II				Dataset III			
	ACC	std	F1-macro	std	ACC	std	F1-macro	std	ACC	std	F1-macro	std
w/o FTL	70.39	6.89	69.89	7.39	70.48	14.76	68.63	13.97	63.31	11.67	61.79	13.25
w/o Dense connection	75.84	7.24	75.40	7.67	70.85	14.11	68.86	14.22	72.46	14.73	69.30	18.88
w/o IP-Unit	72.01	7.26	71.28	8.06	73.28	12.10	70.98	12.64	67.29	11.75	64.48	14.73
EEG-Deformer (ours)	82.72	8.00	82.36	8.52	79.32	7.87	75.83	11.54	73.18	15.63	69.99	19.80

EEG system, with a sampling rate of 500 Hz. The official pre-processed data are used. Following [15], 11 subjects who had enough samples for each class are utilized. The data is down-sampled to 128 Hz. Each sample contains of 3-second EEG signals.

Dataset III: Dataset III provides 19-channel EEG recordings from 36 subjects engaged in mental cognitive tasks, specifically performing serial subtraction, along with their corresponding baseline EEGs for reference. The official artifact-free data are used in this study. Each mental workload trial lasts for 60 seconds, and the last 60 seconds of the rest EEG are used as the low workload data. The trials are segmented using a 4-second sliding window with a moving step of 2 seconds.

B. Baselines

We demonstrate the performance of EEG-Deformer by comparing it with the following baseline methods: 1) two graph-based methods, DGCNN [8] and LGGNet [15]; 2) two CNN-based methods, EEGNet [13] and TSception [14]; 3) four Transformers, EEG-ViT, our adapted version of [5], SSVEPformer [17], EEG-Transformer [28], and EEG-Conformer [6].

1) *DGCNN*: DGCNN employs a novel approach by utilizing a learnable adjacency matrix within its neural network training framework. This matrix dynamically captures and updates the relationships between different EEG channels throughout the training process. By adapting to the evolving data patterns, this method allows for a more flexible and accurate representation of the EEG signal.

2) *LGGNet*: LGGNet is a graph neural network inspired by neurology, designed to capture local-global-graph representations of EEG data. The input layer of LGGNet consists of temporal convolutions with multi-scale 1D convolutional kernels and attentive fusion, effectively capturing the temporal

dynamics of EEG. This data feeds into local and global graph-filtering layers that model complex relationships within and between brain functional areas.

3) *EEGNet*: EEGNet is a streamlined convolutional neural network tailored for EEG-based BCIs. It incorporates depthwise and separable convolutions to develop an EEG-specific model that effectively integrates established EEG feature extraction techniques for BCI applications, making it a compact and effective EEG decoding neural network architecture.

4) *TSception*: TSception is a multi-scale convolutional neural network. The architecture of TSception integrates dynamic temporal layers, asymmetric spatial layers, and advanced fusion layers. This design synergistically extracts discriminative features by leveraging both temporal dynamics and spatial asymmetry across time and channel dimensions. This multi-faceted approach enables TSception to effectively capture the complex patterns essential for accurate emotion recognition from EEG signals.

5) *EEG-ViT*: To explore the effects of combining CNNs and Transformers for decoding EEG signals, we adapt ViT [5], a purely Transformer-based architecture, to EEG data. We partition each EEG input into non-overlapping segments along the temporal dimension and then convert these segments into tokens using a linear projection layer. This allows ViT to effectively learn the temporal relationships among the tokens.

6) *SSVEPformer*: The SSVEPformer is a Transformer-based method that comprises channel combination, SSVEPformer encoder, multilayer perceptron (MLP) head three core components that can extract spectrum information from the complex spectrum representation of EEG.

7) *EEG-Transformer*: The EEG-Transformer is a convolutional Transformer that sequentially integrates CNN and Transformer architectures for EEG signal decoding. This model

utilizes an EEGNet-based CNN feature extractor followed by a vanilla Transformer [23] to effectively process EEG signals.

8) *EEG-Conformer*: It has a similar design principle as EEG-Transformer, which combines CNN and Transformer to learn the spatial-temporal information encoded in EEG signals. EEG-Conformer utilizes one-dimensional CNNs to capture low-level local features and a self-attention module is directly connected to elucidate the global correlations inherent in these local temporal features.

C. Experiment Settings

In this study, we implement generalized subject-independent settings, ensuring that test data information is not used during the training stage. We adopt a leave-one-subject-out (LOSO) approach for all three datasets. In each LOSO step, one subject's data is set aside as test data. Of the remaining data, 80% is used for training and the remaining 20% serves as validation (development) data. We employ Accuracy (ACC) and Macro-F1 as our evaluation metrics which can be formalised as

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (10)$$

$$F1\text{-macro} = \frac{1}{N} \sum_{i=1}^N F1\text{-score}_i, \quad (11)$$

$$F1\text{-score}_i = \frac{2 \cdot TP_i}{2 \cdot TP_i + FP_i + FN_i}, \quad (12)$$

where TP is the true positive, TN is the true negative, FP is the false positive, FN is the false negative, and the subscript indicates the i -th class.

D. Implementation Details

The cross-entropy loss is utilized to guide the training. We use an Adam optimizer with an initial learning rate of 1e-3 and a weight decay of 1e-5. A cosine annealing schedule is adopted to adjust the learning rate during training. A dropout rate of 0.5 is applied on dataset I and II to avoid over-fitting. The dropout rate is changed to 0.25 on dataset III as it achieves a better performance on development set. The batch size is 64 for all the datasets. We train the model for 200 epochs, and the model with the best validation accuracy is used to evaluate the test data. The kernel lengths of CNN layers are calculated by $0.1 * f_s$, where f_s is the sampling rate of the EEG segment. The odd length is needed for Pytorch to achieve the same padding. As the f_s of three datasets are 200 Hz, 128 Hz, and 500 Hz, the kernel lengths of CNN layers $l_{kernel}^{I} = 21$, $l_{kernel}^{II} = 13$, and $l_{kernel}^{III} = 51$. We set the number of CNN kernels as 64 for all the datasets as [15]. We set $n_{head} = d_{atten}$, and tune it based on the performance on the validation (development) set. Hence, we have $n_{head}^I = 32$ and $n_{head}^{II} = n_{head}^{III} = 16$.

V. RESULTS AND ANALYSES

A. Classification Results

Attention Classification: The results for attention classification using dataset I, as presented in Table I, reveal that the EEG-Deformer outperforms all baseline methods. It achieves

the highest accuracy and macro-F1 score in attention classification with 82.72% accuracy and a 82.36% macro-F1 score. Compared to EEG-Conformer, the next best performer, EEG-Deformer shows an improvement of 2.92% in accuracy and 3.35% in macro-F1 score. The results highlight that CNN-based methods generally outperform GNN-based ones and suggest that a combination of CNN and Transformer architectures yields better performance than using either CNN or Transformer alone.

Fatigue Classification: The same observation appears in the fatigue decoding task using dataset II, as detailed in Table I, the EEG-Deformer demonstrates superior performance, leading with an accuracy of 79.32% and a macro-F1 score of 75.83%. These results surpass those of the EEG-Conformer by 4.96% in accuracy and 4.18% in macro-F1 score. The same trend is shown here that the CNN-based methods are better than GNNs. And using Transformer without CNN layers yield low classification results.

Mental Workload Classification: Regarding the classification of mental workload using Dataset III, the EEG-Deformer continues to outperform its counterparts, achieving an accuracy of 73.18% and a macro-F1 score of 69.99%. Unlike in other tasks, the CNN-based method TSception ranks second here, closely followed by EEG-Conformer.

As a summary: 1) RNN-based methods is inferior to CNN-Transformers, EEG-Conformer, as indicated in [6]. Because EEG-Conformer has a better hierarchical sequence learning ability. 2) GNN-based methods have worse performance as they depend heavily on spatial connectivity patterns, which can vary across subjects [29], and are further constrained by EEG's high temporal but low spatial resolution. 3) The results show that Convolutional Transformers excel over traditional Transformers in EEG classification tasks, which is consistent with the observations in [16]. 4) Although the EEG-Conformer outpaces other baselines, our model surpasses it by effectively capturing coarse-to-fine temporal information and utilizing multi-layer insights. Our enhancements specifically address these gaps, advancing EEG data analysis by learning coarse-to-fine temporal information and utilizing purified information from each Transformer layer.

B. Ablation

To investigate the individual contributions of the FTL, Dense connections of IP-Units, and IP-Unit, we conduct an ablation study by selectively removing these layers and observing their impact on classification outcomes across three datasets. The findings, detailed in Table II, indicate that the omission of the FTL had the most detrimental effect on performance overall. Additionally, the removal of the IP-Unit results in the second-largest decrease in performance on datasets I and II. Eliminating dense connections also leads to reductions in both classification accuracy and macro-F1 scores across all datasets. These results collectively imply that each module within the EEG-Deformer plays a crucial role, working in unison to enhance its predictive capabilities.

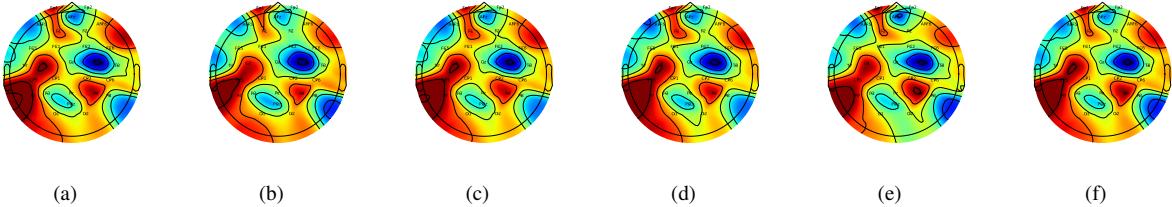


Fig. 5. Saliency maps of attention classification tasks. (a)-(e) are five representative subjects and (f) is the average of all the subjects. The most informative areas are primarily located in the frontal (Fp1, F1, and AFF6) and parietal (CP5, P7, and P4) regions. The location of each EEG electrode can be found according to its name on the saliency map.

TABLE III

EFFECTS OF DIFFERENT INFORMATION PURIFICATION METHODS ON THREE BENCHMARKING DATASETS, WITH THE BEST RESULTS HIGHLIGHTED IN BOLD.

Method	Dataset I				Dataset II				Dataset III			
	ACC	std	F1-macro	std	ACC	std	F1-macro	std	ACC	std	F1-macro	std
Using mean	77.35	6.99	76.99	7.33	73.82	12.13	71.72	12.93	68.82	15.44	65.47	18.90
Using std	77.54	7.54	77.04	8.07	73.94	14.42	72.34	14.77	72.27	15.20	69.19	19.00
Using power	82.72	8.00	82.36	8.52	79.32	7.87	75.83	11.54	73.18	15.63	69.99	19.80

TABLE IV

EFFECTS OF VARIOUS INFORMATION PURIFICATION LOCATIONS ON THREE BENCHMARKING DATASETS, WITH THE BEST RESULTS HIGHLIGHTED IN BOLD.

Method	Dataset I				Dataset II				Dataset III			
	ACC	std	F1-macro	std	ACC	std	F1-macro	std	ACC	std	F1-macro	std
At \mathbf{F}_{i+1}	80.28	7.46	79.81	8.03	77.25	7.14	73.98	9.96	69.16	15.46	66.35	18.19
At \mathbf{F}_{cg}^i	72.90	7.30	72.05	8.35	44.72	16.48	29.96	8.30	71.41	15.30	68.56	18.50
At \mathbf{F}_{fg}^i	82.72	8.00	82.36	8.52	79.32	7.87	75.83	11.54	73.18	15.63	69.99	19.80

C. Ablation of IP-Units

To further understand the role of each IP-Unit, we conducted a removal analysis where each IP-Unit was individually omitted to observe its impact. Figure 6 depicts the consequences of this removal, displaying reductions in accuracy (ACC) and F1 score. The results show that eliminating the first IP-Unit resulted in the largest declines on datasets I and II, whereas the third IP-Unit was more crucial for dataset III. These findings seem to correlate with the lengths of the input EEG segments, which are 800, 384, and 2000 for datasets I, II, and III, respectively. This suggests that the importance of deeper IP-Units increases with the length of the input EEG segment. Additionally, removing the last IP-Unit also led to significant reductions in both ACC and macro-F1 on datasets I and II, underscoring that all IP-Units collectively contribute to performance enhancements.

D. Effect of Different Types of IP-Unit

Table III illustrates the impact of different information purification methods, which calculate mean, standard deviation (std), and power along the feature dimension of F_{fg}^i , on the performance across three datasets. The comparison focuses on their effects on accuracy and macro-F1 score. Notably, the 'power' method consistently yields the highest accuracy and macro-F1 scores across all datasets, outperforming the other two methods. These findings underscore the superiority

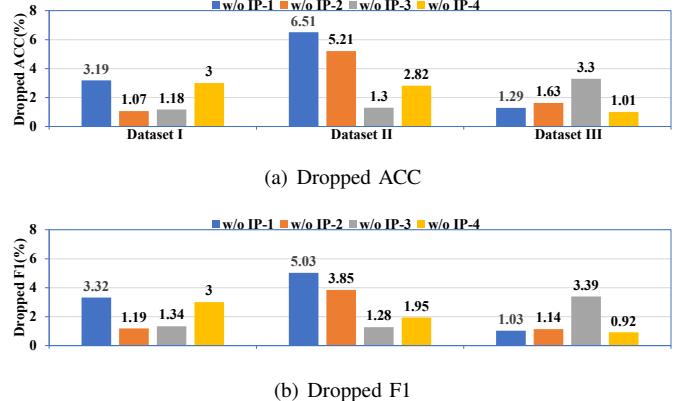


Fig. 6. Dropped ACC (a) and F1 scores (b) resulted from removing each IP-Unit from EEG-Deformer. Removing the first IP-Unit significantly impacted performance on datasets I and II, while the third IP-Unit was vital for dataset III, reflecting their input EEG segment lengths of 800, 384, and 2000, respectively. This indicates that longer EEG inputs require deeper IP-Units.

of using the 'power' method for information purification in this context.

E. Effect of Different Locations of The IP-Unit

The study assesses the impact of varying the placement of IP-Unit within each HCT block, focusing on three specific locations in Figure 4 : at the fused output, \mathbf{F}_{i+1} ; at the

coarse-grained representation, F_{cg}^i ; and at the fine-grained representation, F_{fg}^i . The results of this evaluation are presented in Table IV. These findings reveal that applying IP-Unit to the output of the FTL leads to the highest decoding performance across all three datasets. This outcome highlights the effectiveness of this particular placement for IP-Unit within the HCT block structure.

F. The Parameters Comparison of EEG-Deformer and Other Baseline Methods.

We visualize the comparison between different methods with respect to parameters, Multiply-Accumulate Operations (MACs), and accuracy. The results are shown in Fig. 7. We only compare models that use EEG as input, as the hand-crafted features have fewer data points in each input. On dataset I, the EEG-Deformer requires fewer MACs than models of similar size, while achieving higher accuracy than smaller models. On dataset II, the EEG-Deformer has the highest performance with a relatively smaller model size and fewer MACs compared to other baselines. On dataset III, although the EEG-Deformer uses more MACs, it has a smaller model size than those requiring more MACs. Generally, our EEG-Deformer achieves a promising balance between accuracy and computational efficiency.

G. Visualization

In this study, saliency maps [30] are utilized to visualize the areas deemed most informative by the neural networks. Figures 5 to 9 display saliency maps for five representative subjects and the average of all subjects across three datasets, corresponding to attention, fatigue, and mental workload classification tasks. For enhanced visualization, these saliency maps are normalized to values between 0 and 1. The human brain can be divided into frontal, temporal, parietal, and occipital functional areas. According to Figure 5, the areas most informative for attention classification are primarily located in the frontal (Fp1, F1, and AFF6) and parietal (CP5, P7, and P4) regions, which are known to be related to cognitive attention [31]. Figure 8 shows that for fatigue classification, the neural network primarily focuses on the frontal (F7, F3, and FCz), temporal (T3 and T5), and parietal (P3) areas, consistent with [32]. In the case of mental workload classification, we find consistent findings with those reported in [33]: the frontal (Fz and Fp2) and parietal (P4) areas are more informative for neural networks, as shown in Figure 9. These visualizations align with known brain activity patterns related to these cognitive processes and demonstrate the neural network's ability to identify relevant brain regions for each task.

VI. CONCLUSION

In this paper, we introduce EEG-Deformer, a novel convolutional Transformer designed for cross-subject EEG classification tasks. The model begins with a shallow CNN encoder, followed by a series of HCT blocks. These blocks are specifically designed to capture both coarse- and fine-grained temporal dynamics from EEG signals, achieved by

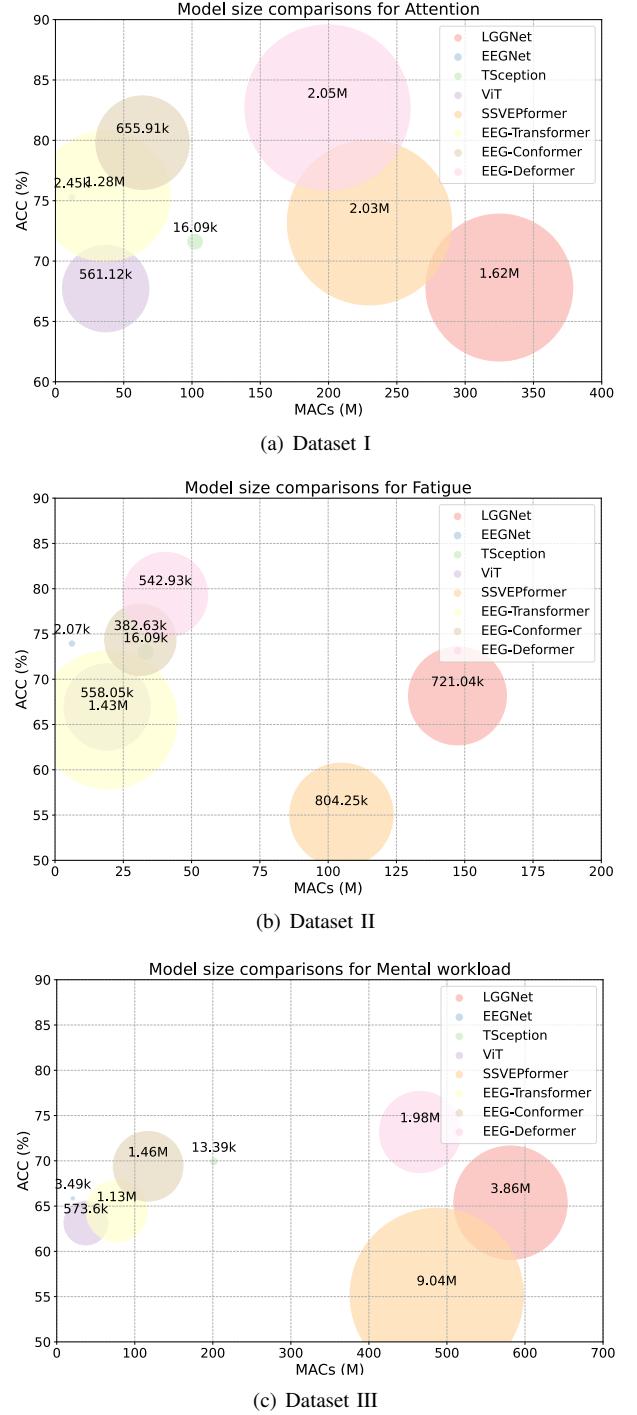


Fig. 7. Comparisons about performance (ACC, on y-axis), model size (size of bubbles), and computational complexity (MACs, on x-axis). Light pink one is Deformer (ours). Our EEG-Deformer strikes a promising balance between accuracy and computational complexity.

integrating a FTL into the Transformers. To leverage multi-level temporal information effectively, the model performs information purification on the fine-grained temporal representations extracted by all HCT layers. These purified signals are then densely connected to the final embeddings. We evaluate EEG-Deformer against various baselines across three benchmark datasets. The results consistently demonstrate that EEG-Deformer outperforms these baselines, showcasing its ef-

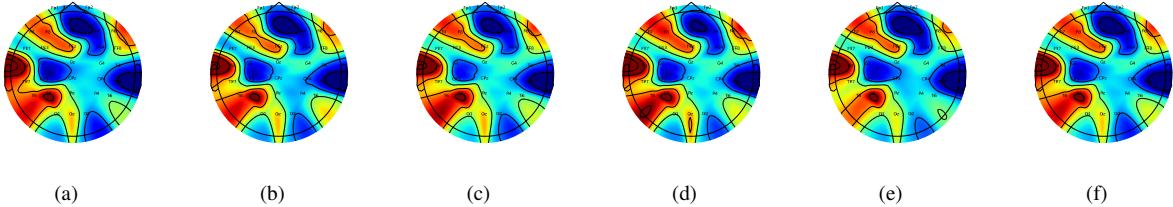


Fig. 8. Saliency maps of fatigue classification tasks are shown. Figures (a) to (e) represent five representative subjects, and figure (f) is the average of all the subjects. The most informative areas are the frontal (F7, F3, and FCz), temporal (T3 and T5), and parietal (P3) regions. The location of each EEG electrode can be identified by its name on the saliency map.

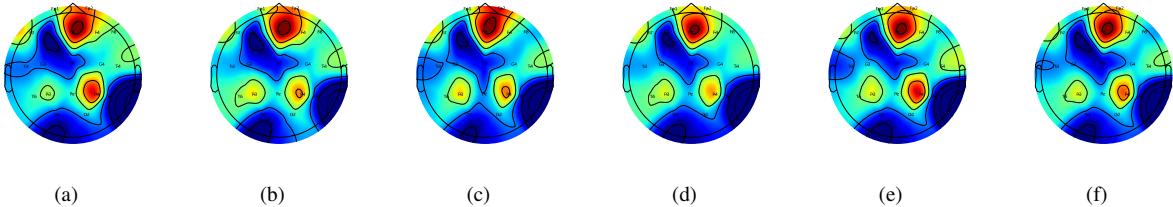


Fig. 9. Saliency maps for mental workload classification tasks are presented. Figures (a) to (e) represent five representative subjects, and figure (f) shows the average of all subjects. The frontal (Fz and Fp2) and parietal (P4) areas are found to be more informative. The location of each EEG electrode can be identified by its name on the saliency map.

effectiveness in EEG decoding tasks. These findings suggest that EEG-Deformer can serve as a robust and versatile backbone for a wide range of EEG decoding tasks.

ACKNOWLEDGMENT

This work was supported by the RIE2020 AME Programmatic Fund, Singapore (No. A20G8b0102) and was partially supported by the MoE AcRF Tier 1 Project (No. RT01/21).

REFERENCES

- [1] X. Zhang, J. Liu, J. Shen, S. Li, K. Hou, B. Hu, J. Gao, T. Zhang, and B. Hu, "Emotion recognition from multimodal physiological signals using a regularized deep fusion of kernel machine," *IEEE Transactions on Cybernetics*, pp. 1–14, 2020.
- [2] F. Lotte and C. Guan, "Regularizing common spatial patterns to improve BCI designs: unified theory and new algorithms," *IEEE Transactions on biomedical Engineering*, vol. 58, no. 2, pp. 355–362, 2010.
- [3] R. Foong, K. K. Ang, C. Quek, C. Guan, K. S. Phua, C. W. K. Kuah, V. A. Deshmukh, L. H. L. Yam, D. K. Rajeswaran, N. Tang *et al.*, "Assessment of the efficacy of EEG-based MI-BCI with visual feedback and EEG correlates of mental fatigue for upper-limb stroke rehabilitation," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 3, pp. 786–795, 2019.
- [4] V. Zotev, A. Mayeli, M. Misaki, and J. Bodurka, "Emotion self-regulation training in major depressive disorder using simultaneous real-time fMRI and EEG neurofeedback," *NeuroImage: Clinical*, vol. 27, p. 102331, 2020.
- [5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uz/Dkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.
- [6] Y. Song, Q. Zheng, B. Liu, and X. Gao, "EEG Conformer: Convolutional transformer for EEG decoding and visualization," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 31, pp. 710–719, 2023.
- [7] P. Autthasan, R. Chaisaen, T. Sudhawiyangkul, P. Rangpong, S. Kitathaveephong, N. Dilokthanakul, G. Bhakdisongkhram, H. Phan, C. Guan, and T. Wilairasitporn, "Min2net: End-to-end multi-task learning for subject-independent motor imagery eeg classification," *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 6, pp. 2105–2118, 2022.
- [8] T. Song, W. Zheng, P. Song, and Z. Cui, "EEG emotion recognition using dynamical graph convolutional neural networks," *IEEE Transactions on Affective Computing*, vol. 11, no. 3, pp. 532–541, 2020.
- [9] V. Delvigne, H. Wannous, T. Dutoit, L. Ris, and J.-P. Vandeborre, "Phydaa: Physiological dataset assessing attention," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 2612–2623, 2022.
- [10] Y. Ding and C. Guan, "GIGN: Learning graph-in-graph representations of EEG signals for continuous emotion recognition," in *2023 45th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2023, pp. 1–5.
- [11] Y. Ding, S. Zhang, C. Tang, and C. Guan, "MASA-TCN: Multi-anchor space-aware temporal convolutional neural networks for continuous and discrete EEG emotion recognition," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–12, 2024.
- [12] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball, "Deep learning with convolutional neural networks for EEG decoding and visualization," *Human Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, 2017.
- [13] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: a compact convolutional neural network for EEG-based brain-computer interfaces," *Journal of Neural Engineering*, vol. 15, no. 5, p. 056013, Jul 2018.
- [14] Y. Ding, N. Robinson, S. Zhang, Q. Zeng, and C. Guan, "TSception: Capturing temporal dynamics and spatial asymmetry from EEG for emotion recognition," *IEEE Transactions on Affective Computing*, vol. 14, no. 3, pp. 2238–2250, 2023.
- [15] Y. Ding, N. Robinson, C. Tong, Q. Zeng, and C. Guan, "LGGNet: Learning from local-global-graph representations for brain–computer interface," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2023.
- [16] J. Xie, J. Zhang, J. Sun, Z. Ma, L. Qin, G. Li, H. Zhou, and Y. Zhan, "A transformer-based approach combining deep learning network and spatial-temporal information for raw EEG classification," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 30, pp. 2126–2136, 2022.
- [17] J. Chen, Y. Zhang, Y. Pan, P. Xu, and C. Guan, "A transformer-based deep neural network model for ssvep classification," *Neural Networks*, vol. 164, pp. 521–534, 2023.
- [18] Y.-E. Lee and S.-H. Lee, "EEG-Transformer: Self-attention from transformer architecture for decoding EEG of imagined speech," in *2022 10th International Winter Conference on Brain-Computer Interface (BCI)*, 2022, pp. 1–4.
- [19] A. Sikka, H. Jamalabadi, M. Krylova, S. Alizadeh, J. N. van der

- Meer, L. Danyeli, M. Deliano, P. Vicheva, T. Hahn, T. Koenig, D. R. Bathula, and M. Walter, "Investigating the temporal dynamics of electroencephalogram (eeg) microstates using recurrent neural networks," *Human Brain Mapping*, vol. 41, no. 9, pp. 2334–2346, 2020. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/hbm.24949>
- [20] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2261–2269.
- [21] P. L. Nunez and R. Srinivasan, *Electric fields of the brain: the neuro-physics of EEG*. Oxford University Press, USA, 2006.
- [22] R. Mane, N. Robinson, A. P. Vinod, S.-W. Lee, and C. Guan, "A multi-view cnn with novel variance layer for motor imagery brain computer interface," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2020, pp. 2950–2953.
- [23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.
- [24] T.-P. Jung, S. Makeig, M. Stensmo, and T. Sejnowski, "Estimating alertness from the eeg power spectrum," *IEEE Transactions on Biomedical Engineering*, vol. 44, no. 1, pp. 60–69, 1997.
- [25] J. Shin, A. von Lüthmann, D.-W. Kim, J. Mehnert, H.-J. Hwang, and K.-R. Müller, "Simultaneous acquisition of EEG and NIRS during cognitive tasks for an open access dataset," *Scientific Data*, vol. 5, no. 1, p. 180003, 2018.
- [26] Z. Cao, C.-H. Chuang, J.-K. King, and C.-T. Lin, "Multi-channel EEG recordings during a sustained-attention driving task," *Scientific data*, vol. 6, no. 1, pp. 1–8, 2019.
- [27] I. Zyma, S. Tukaev, I. Seleznov, K. Kiyono, A. Popov, M. Chernykh, and O. Shpenkov, "Electroencephalograms during mental arithmetic task performance," *Data*, vol. 4, no. 1, 2019.
- [28] Y.-E. Lee and S.-H. Lee, "Eeg-transformer: Self-attention from transformer architecture for decoding eeg of imagined speech," in *2022 10th International winter conference on brain-computer interface (BCI)*. IEEE, 2022, pp. 1–4.
- [29] S. H. Tompson, E. B. Falk, J. M. Vettel, and D. S. Bassett, "Network approaches to understand individual differences in brain connectivity: opportunities for personality neuroscience," *Personality neuroscience*, vol. 1, p. e5, 2018.
- [30] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint arXiv:1312.6034*, 2013.
- [31] Y. Liu, J. Bengson, H. Huang, G. R. Mangun, and M. Ding, "Top-down Modulation of Neural Activity in Anticipatory Visual Attention: Control Mechanisms Revealed by Simultaneous EEG-fMRI," *Cerebral Cortex*, vol. 26, no. 2, pp. 517–529, 09 2014.
- [32] P. Flor-Henry, J. C. Lind, and Z. J. Koles, "EEG source analysis of chronic fatigue syndrome," *Psychiatry Research: Neuroimaging*, vol. 181, no. 2, pp. 155–164, 2010.
- [33] I. Kakkos, G. N. Dimitrakopoulos, Y. Sun, J. Yuan, G. K. Matsopoulos, A. Bezerianos, and Y. Sun, "EEG fingerprints of task-independent mental workload discrimination," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 10, pp. 3824–3833, 2021.

EEG2TEXT: Open Vocabulary EEG-to-Text Decoding with EEG Pre-Training and Multi-View Transformer

¹Hanwen Liu, ¹Daniel Hajialigol, ²Benny Antony, ³Aiguo Han, ¹Xuan Wang

¹ Department of Computer Science, Virginia Tech, VA, USA

² Department of Electrical and Computer Engineering, Virginia Tech, VA, USA

³ Department of Biomedical Engineering and Mechanics, Virginia Tech, VA, USA

{liuhwen, danielhajialigol, bennyantony, aiguohan, xuanw}@vt.edu

Abstract

Deciphering the intricacies of the human brain has captivated curiosity for centuries. Recent strides in Brain-Computer Interface (BCI) technology, particularly using motor imagery, have restored motor functions such as reaching, grasping, and walking in paralyzed individuals. However, unraveling natural language from brain signals remains a formidable challenge. Electroencephalography (EEG) is a non-invasive technique used to record electrical activity in the brain by placing electrodes on the scalp. Previous studies of EEG-to-text decoding have achieved high accuracy on small closed vocabularies, but still fall short of high accuracy when dealing with large open vocabularies. We propose a novel method, EEG2TEXT, to improve the accuracy of open vocabulary EEG-to-text decoding. Specifically, EEG2TEXT leverages EEG pre-training to enhance the learning of semantics from EEG signals and proposes a multi-view transformer to model the EEG signal processing by different spatial regions of the brain. Experiments show that EEG2TEXT has superior performance, outperforming the state-of-the-art baseline methods by a large margin of up to 5% in absolute BLEU and ROUGE scores. EEG2TEXT shows great potential for a high-performance open-vocabulary brain-to-text system to facilitate communication.

1 Introduction

Recent advances in brain-computer interface (BCI) technology have demonstrated exciting progress in restoring the capabilities of patients with paralysis, particularly restoring such motor functions as reaching [Hochberg *et al.*, 2012], grasping [Aflalo *et al.*, 2015; Bouton *et al.*, 2016], and walking [Lorach *et al.*, 2023]. The heart of BCI is its ability to accurately decode complex brain signals. Despite the advances in decoding brain signals related to motion, decoding brain signals related to speech remains a formidable challenge. Previous research translating speech-related brain signals to text (brain-to-text) primarily relies on electrocorticography (ECoG), an invasive electrophysiological monitoring method

that uses electrodes placed directly on the exposed brain surface to record activity from the cerebral cortex. ECoG offers higher temporal and spatial resolution than traditional non-invasive scalp electroencephalography (EEG), with a significantly better signal-to-noise ratio. However, the invasive nature of ECoG is undesirable for BCI applications, and it is highly desirable to develop brain-to-text decoding methods using noninvasive EEG signals, although EEG signals are significantly more challenging to work with than ECoG.

Previous studies of EEG-to-text decoding [Herff *et al.*, 2015; Sun *et al.*, 2019; Anumanchipalli *et al.*, 2019; Makin *et al.*, 2020; Panachakel and Ramakrishnan, 2021; Moses *et al.*, 2021; Nieto *et al.*, 2022] have achieved high accuracy on small closed vocabularies, but still fall short of high accuracy when dealing with large open vocabularies. These approaches primarily target high accuracy (> 90%) but are often confined to small closed vocabularies and struggle to decode semantically similar words beyond training sets. Recent studies broaden the scope from closed to open-vocabulary EEG-to-text decoding [Wang and Ji, 2021; Willett *et al.*, 2023; Tang *et al.*, 2023; Duan *et al.*, 2023], drastically expanding the vocabulary size by over 100-fold, from several hundred to tens of thousands of words. Notably, two of these studies [Wang and Ji, 2021; Duan *et al.*, 2023] leverage a pre-trained large language model BART [Lewis *et al.*, 2019], and represent the state-of-the-art for open vocabulary brain-to-text decoding. However, these studies are in their nascent stages and are challenged by their limited accuracy.

To improve the accuracy of EEG-to-text decoding with open vocabularies, we propose a novel EEG-to-text decoding method based on transformers. First, we introduce a Convolutional Neural Network (CNN) module before the base transformer model to enhance the model’s ability to handle long EEG signals. Second, we conduct pre-training of the transformer model by reconstructing randomly masked EEG signals from the input data. This pre-training step helps our transformer model better learn the semantics of EEG signals. Last, we propose a multi-view transformer architecture, where each single-view transformer is the pre-trained model from the previous step, to model the EEG signal processing by different spatial regions of the brain. Experiments show that EEG2TEXT has superior performance, outperforming the state-of-the-art baseline methods by a large margin of up to 5% in absolute BLEU and ROUGE scores.

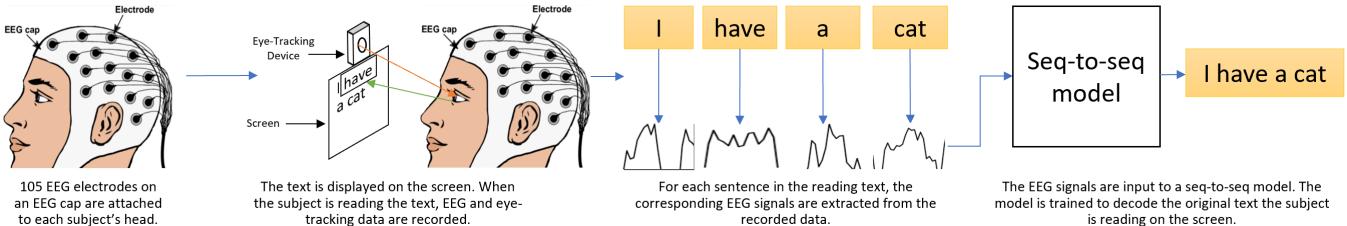


Figure 1: The overall framework of open-vocabulary EEG-to-text translation. The first sub-figure comes from [Nagel and Spüler, 2018].

EEG2TEXT shows great potential for a high-performance open-vocabulary brain-to-text system to facilitate communication. We will open-source our code and dataset to facilitate future studies of EEG-to-text translation.

2 Task Definition

Our task involves decoding corresponding natural language text from raw EEG signals (Figure 1). The data acquisition process involves 1) attaching an EEG cap to each subject’s head, 2) displaying the text (reading materials) on a screen, and 3) recording the EEG and eye-tracking (for verification and calibration of the EEG signals) data while the subject is reading the text. The EEG signals are further extracted from the recorded data and fed as input to a decoding model to predict the original text the subject was reading on the screen.

Formally, this task can be formulated as a sequence-to-sequence machine translation task as follows:

$$P(Y|X) = \arg \max_Y \prod_{t=1}^{T'} P(y_t|y_{<t}, X) \quad (1)$$

where T' represents the length of the target sentence Y ; y_t represents the word or token at position t in the target sentence Y ; $y_{<t}$ represents the words or tokens preceding position t in the target sentence Y ; X represents the input EEG data; and $P(y_t|y_{<t}, X)$ is the conditional probability of generating word y_t given the previous words $y_{<t}$ and the input EEG data X . Our goal is to maximize the probability $P(Y|X)$ of generating the target sentence given the input EEG data.

3 Methodology

3.1 Baseline Model

Our baseline model [Wang and Ji, 2021] takes the word-level EEG features as the input to a transformer model followed by a pre-trained BART model for text decoding. The raw EEG signals are typically stored as a two-dimensional array with one dimension for time and the other for channels (the number of electrodes used to collect EEG signals). Each value in this two-dimensional array corresponds to the signal strength collected at the corresponding time for the corresponding channel. In the baseline model, the word-level EEG features are extracted from eight independent frequency bands from the raw EEG signals. The above eight word-level EEG features are simply concatenated across all the channels as input to the decoder framework.

The baseline model faces the following challenges: 1) the reliance on eye-tracking calibration for word-level EEG feature extraction introduces error propagation and lacks generalizability to scenarios such as inner speech decoding [Martin *et al.*, 2018; Nalborczyk *et al.*, 2020], 2) there is room for improvement in EEG representation learning through self-supervised pre-training, and 3) the lack of spatial resolution modeling ignores the varying importance of different brain regions in language processing. To overcome these challenges, we propose a novel framework, EEG2TEXT, that achieves superior performance for open-vocabulary EEG-to-text translation.

3.2 Convolutional Transformer for Sentence-Level EEG Encoding

Instead of using the word-level EEG features crafted based on the eye-tracking data, we directly use the sentence-level EEG signals as input to our model. Using sentence-level EEG signals offers several advantages over word-level EEG features. It provides richer information without error propagation from the eye-tracking data and exhibits better generalizability to other tasks, such as inner speech decoding, where acquiring eye-tracking data is infeasible.

However, the sentence-level EEG signals pose a challenge due to their excessive length, potentially overloading laboratory-level GPUs if directly input into the transformer layer. To tackle this issue, we introduce a convolutional transformer model that incorporates a CNN module for compressing raw EEG signals. This CNN module comprises two convolutional layers, adept at both temporal and spatial (or channel) compression. We also compared two input formats of the sentence-level EEG signals: 1) the raw signals, and 2) the spectrogram of the signals. The spectrogram of a signal (Appendix Figure A1) is a two-dimensional image, where the x-axis represents time, the y-axis represents frequency, and the image pixel value represents the magnitude of the signal at each time-frequency pair. The sentence-level EEG signals are then input into the CNN module to obtain compressed EEG signals, which are then fed into the transformer model for subsequent feature extraction and text translation.

3.3 Transformer Pre-Training for an Enhanced EEG Encoding

To enhance the semantic understanding of the EEG signals, we propose a self-supervised pre-training of the convolutional transformer model for parameter initialization (Figure 2). Inspired by the masked language model pre-training strategies [Devlin *et al.*, 2018; Joshi *et al.*, 2019; Liu *et al.*, 2019], we

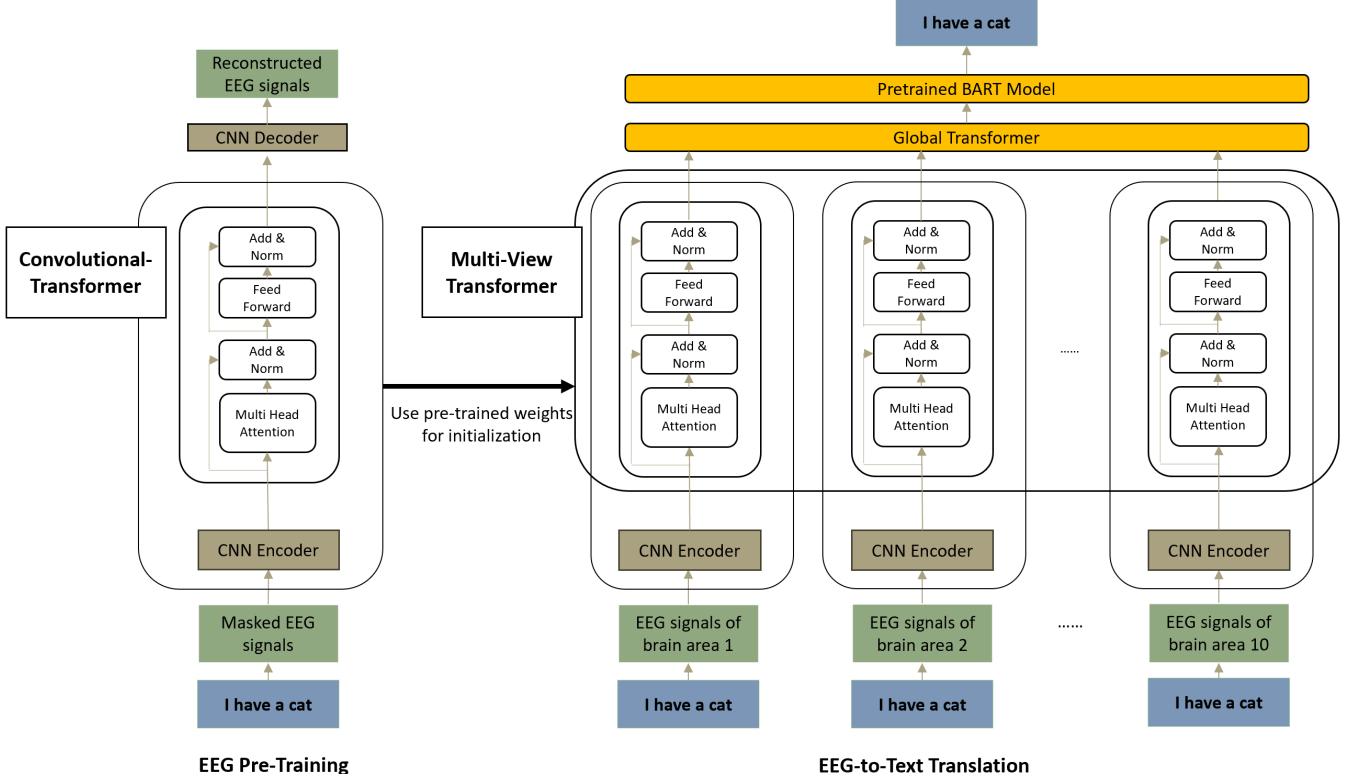


Figure 2: The overall framework of EEG2TEXT. It takes the sentence EEG signals as input and decodes the original text as output. EEG2TEXT includes major steps of 1) a base convolutional transformer model, 2) pre-training for EEG encoding, and 3) a multi-view transformer for different spatial regions of the brain.

formulate our self-supervised pre-training objective as follows:

$$\theta^* = \arg \max_{\theta} \sum_{(i,j) \in \mathcal{D}} \log P(M|C; \theta), \quad (2)$$

where M represents the masked tokens; C represents the context or surrounding tokens; θ^* represents the optimal model parameters; θ represents the model parameters being optimized; \mathcal{D} represents the training data, where (i, j) are pairs of sentences or sentence fragments; and $P(M|C; \theta)$ is the probability of predicting the masked tokens.

During the self-supervised pre-training stage, we add a convolutional decoder module on top of the convolutional transformer encoder to decode the input EEG signals. The input is the sentence-level EEG signals masked with different strategies and the output is the sentence-level EEG signals reconstructed by the CNN decoder. Specifically, we compared three different masking strategies for the sentence-level EEG signals as follows:

- **Masked Token Prediction** [Devlin *et al.*, 2018]: randomly masking 15% of all the tokens.
- **Continuous Masked Token Prediction** [Joshi *et al.*, 2019]: randomly masking a sequence of consecutive tokens until a total of 15% of all the tokens are masked.
- **Re-Masked Token Prediction** [Liu *et al.*, 2019]: re-randomizing the masking of 15% of all the tokens for each training epoch.

It is important to highlight that our self-supervised pre-training step allows for seamless integration of EEG data from diverse tasks, including image recognition. In our experiments, we further incorporated an image EEG dataset [Gifford *et al.*, 2022] during pre-training, aiming to showcase the model’s adaptability to EEG signals from multi-modal data and explore the potential for enhanced translation performance through the combination of EEG signals from diverse data modalities.

The goal of this pre-training step is to have the convolutional transformer learn meaningful concepts such as context, relationships, and semantics present in sentence-level EEG signals during this pre-training process. After pre-training, the parameters are saved and used as the initial parameters for the final multi-view transformer model.

3.4 Multi-View Transformer for Different Spatial Regions of the Brain

Another important feature of our model is the novel multi-view transformer decoder architecture we introduced that encodes different regions of the brain with a different convolutional transformer (Figure 2). The multi-view transformer model takes into account the fact that different brain regions potentially play different roles in language processing. This spatial modeling therefore can improve the model performance, but has been overlooked in previous work.

Approximate Brain Areas	Corresponding Electrodes
Prefrontal Cortex	E6, E12, E5, E11, E16, E15, E20, E118, E24, E124, E26, E2, E27, E123, E3, E4, E23, E19, E22, E9, E10, E18, E28, E33, E117, E122
Premotor Cortex	CZ, E7, E106, E105, E104, E115, E114, E120, E110, E116, E121, E111, E112, E109, E13, E30
Broca’s Area	E29, E36, E35, E34
Auditory Association Area	E40, E38, E39, E43, E44, E46, E57, E58, E64
Primary Motor Cortex	E31, E80, E55, E37, E87, E93, E103, E102, E108
Primary Sensory Cortex	E54, E79, E61, E78, E62, E53, E86, E92, E98, E100, E101
Somatic Sensory Cortex	E67, E77, E71, E72, E76, E66, E84, E60, E85
Auditory Cortex	E59, E91, E97, E51
Wernicke’s Area	E41, E42, E52, E47, E45, E50
Visual Area	E65, E69, E70, E74, E75, E82, E83, E89, E90, E95, E96

Table 1: Ten channel groups and their corresponding approximate brain areas.

We partition the 105 channels into ten groups based on their spatial location under the guidance of functional brain regions (Table 1). Specifically, we compared the spatial distribution of 105 electrodes with the spatial distribution of functional brain regions and mapped each electrode to its closest brain region. Details of the electrode spatial distribution can be found in [Hollenstein *et al.*, 2018].

After the partition of the electrodes, we create a multi-view transformer model including ten convolutional transformers at the bottom level, where each convolutional transformer encodes the EEG signals from the electrodes in that region. On top of the ten convolutional transformers, we add a global transformer to unify the information from different brain regions. The combined information from the global transformer is further fed into the BART model for text decoding.

In summary, the multi-view transformer envisions multiple parallel convolutional transformer models where each captures different aspects of EEG signals combined from different spatial regions of the brain regions. This approach enhances the spatial resolution of the model and further improves the text decoding performance.

4 Experiment

4.1 Experimental Setup

Dataset We utilize both the Zuco [Hollenstein *et al.*, 2018] and Image-EEG [Gifford *et al.*, 2022] datasets for pre-training and use Zuco to train the multi-view transformer and BART model for text decoding. Details of both datasets are listed below.

Methods	Batch Size	Learning Rate
EEG2TEXT (Convolutional Transformer)	4	1×10^{-5}
EEG2TEXT (+ Pre-training)	4	5×10^{-5}
EEG2TEXT (+ Multi-View Transformer)	4	3×10^{-5}

Table 2: Optimal hyper-parameters for EEG2TEXT ablations.

- **Zuco** [Hollenstein *et al.*, 2018] contains EEG and eye-tracking data from 12 healthy adult native English speakers engaged in natural English text reading for 4 - 6 hours. This dataset covers two standard reading tasks and a task-specific reading task, offering EEG and eye-tracking data for 21,629 words across 1,107 sentences and 154,173 fixations. We pre-process the data to extract both word-level and sentence-level features to serve as input to our model.
- **Image-EEG** [Gifford *et al.*, 2022] is a large and rich dataset containing high temporal resolution EEG signals of images of objects on natural backgrounds. The dataset included 10 participants, each performing 82,160 trials across 16,740 image conditions.

Baselines We compare EEG2TEXT with two baseline models for open-vocabulary EEG-to-text translation.

- **Baseline** [Wang and Ji, 2021] uses word-level EEG signals as input to a transformer model followed by a pre-trained BART model for decoding.
- **DeWave** [Duan *et al.*, 2023] introduces a discrete codex encoding after the transformer layer, and uses both word-level EEG features and the raw EEG signals as input.

Evaluation Metrics We utilize BLEU-1, BLEU-2, BLEU-3, BLEU-4, and ROUGE-1 evaluation metrics to compare the performance of EEG2TEXT with the baselines.

The BLEU-N scores ($N = 1, 2, 3, 4$) are used to measure the quality of the generated text, with higher values indicating better performance.

$$\text{BLEU} = \text{BP} \cdot \exp \left(\sum_{n=1}^N w_n \cdot \log \left(\frac{\text{count}_{\text{clip},n}}{\text{count}_{\text{ref},n}} \right) \right), \quad (3)$$

where BLEU represents the BLEU score; BP represents the brevity penalty; N represents the max n-gram order; w_n represents the n-gram weights; $\text{count}_{\text{clip},n}$ represents count of candidate n-grams in reference and $\text{count}_{\text{ref},n}$ represents count of reference n-grams.

ROUGE-1 scores, which include F (F1-score), P (precision), and R (recall), are used to evaluate the overlap between generated text and reference text.

$$\text{ROUGE-1} = \frac{\sum_{\text{ref}} \sum_{1\text{-gram}} \min(\text{match}, \text{ref})}{\sum_{\text{ref}} \sum_{1\text{-gram}} \text{ref}}, \quad (4)$$

where ROUGE-1 represents the ROUGE-1 score; match represents the count of matching 1-gram; ref represents the count of 1-gram.

Parameter Study We used four A40 GPUs as our computing infrastructure and each training epoch took about 40 minutes. The optimal hyper-parameters for our results are listed

Methods	BLEU-N				ROUGE-1		
	N = 1	N = 2	N = 3	N = 4	F	P	R
Baseline [Wang and Ji, 2021]	0.401	0.231	0.125	0.068	0.301	0.317	0.288
DeWave [Duan <i>et al.</i> , 2023]	0.413	0.241	0.139	0.082	0.288	0.337	0.306
EEG2TEXT (Convolutional Transformer)	0.400	0.236	0.137	0.082	0.325	0.361	0.297
EEG2TEXT (+ Pre-training)	0.445	0.274	0.175	0.117	0.341	0.383	0.310
EEG2TEXT (+ Multi-View Transformer)	0.452	0.291	0.197	0.141	0.342	0.369	0.320

Table 3: Performance comparison of EEG2TEXT with baseline methods.

Methods	BLEU-N				ROUGE-1		
	N = 1	N = 2	N = 3	N = 4	F	P	R
Spectrogram + Transformer	0.386	0.220	0.121	0.067	0.306	0.342	0.306
Spectrogram + Convolutional Transformer	0.374	0.209	0.112	0.061	0.302	0.339	0.274
EEG signal + Convolutional Transformer	0.400	0.236	0.137	0.082	0.325	0.361	0.297

Table 4: Ablation study of different input formats of the EEG signals.

in Table 2. The value ranges of each hyper-parameter are listed below:

- Batch Size $\in \{4, 8, 16\}$
- Learning Rate $\in \{1 \times 10^{-6}, 3 \times 10^{-6}, 5 \times 10^{-6}, 7.5 \times 10^{-6}, 8 \times 10^{-6}, 9 \times 10^{-6}, 1 \times 10^{-5}, 2 \times 10^{-5}, 3 \times 10^{-5}, 4 \times 10^{-5}, 5 \times 10^{-5}, 7.5 \times 10^{-5}, 1 \times 10^{-4}, 3 \times 10^{-4}, 5 \times 10^{-4}, 7.5 \times 10^{-4}, 1 \times 10^{-3}\}$

4.2 Results

Main Results Table 3 shows our main experimental results. The baseline method [Wang and Ji, 2021] achieves a moderate performance in text decoding with BLEU scores. DeWave [Duan *et al.*, 2023] slightly improved the performance across all metrics, demonstrating the effectiveness of discrete encoding. EEG2TEXT improved the text decoding performance by a large margin due to several technical innovations. First, a single convolutional transformer achieved slightly lower BLEU scores (BLEU-1: -1.3%; BLEU-2: -0.5%; BLEU-3: -0.2%; BLEU-4: -0.0%) but higher ROUGE-1 scores (F1-score: +3.7%; Precision: +2.4%; Recall: -0.9%) compared to DeWave. Second, EEG2TEXT with pre-training further enhanced the BLEU scores (BLEU-1: +1.8%; BLEU-2: +1.9%; BLEU-3: +1.8%; BLEU-4: +1.6%) and ROUGE-1 scores (F1-score: +4.2%; Precision: +2.4%; Recall: +0.0%) compared to DeWave. Pre-training proved effective in enhancing text generation by providing a strong initialization foundation for our model. Third, EEG2TEXT with multi-view transformers achieved the highest scores across all metrics, with a significant increase in the BLEU scores (BLEU-1: **+3.9%**; BLEU-2: **+5.0%**; BLEU-3: **+5.8%**; BLEU-4: **+5.9%**) and ROUGE-1 scores (F1-score: **+5.4%**; Precision: **+3.2%**; Recall: **+1.4%**) compared to DeWave. EEG2TEXT excelled in generating coherent, contextually relevant, and high-quality text.

Convolutional Transformer We first compare different input representations of the EEG signals to see how the representation affects the performance of a base convolutional

transformer model. In this ablation study, we compare the raw EEG signals with their spectrograms using the fast Fourier transform [Cochran *et al.*, 1967] to convert the original one-dimensional time array into a two-dimensional time-frequency matrix. The results are shown in Table 4. Using the raw EEG as the input consistently led to better performance than using the spectrogram as the input. Because the spectrogram only keeps the magnitude information and ignores the phase information of the raw EEG signal, the superior performance of the raw EEG signal suggested that the phase information might be important for decoding. Therefore, the raw EEG signals are used as the input in our subsequent experiments.

EEG Pre-Training We then conducted ablation experiments to compare the effectiveness of three pre-training strategies: 1) Masked Token Prediction [Devlin *et al.*, 2018], 2) Continuous Masked Token Prediction, and 3) Re-Masked Token Prediction [Liu *et al.*, 2019]. The results are shown in Table 5. The Re-Masked Token Prediction [Liu *et al.*, 2019] exhibits the best performance among all the three masking strategies. One potential reason is that the convolutional transformer model can learn more diverse semantic information by masking different tokens in each training epoch during pre-training.

In the above study, we focused on identifying the optimal pre-training strategy among the three without incorporating image-EEG data [Gifford *et al.*, 2022]. As an additional component, we introduced image-EEG data to assess the compatibility of our model with EEG signals from multi-modal inputs. Leveraging our self-supervised pre-training strategy, we directly incorporated image-EEG data into the pre-training phase to enable the model to glean knowledge from diverse sources. The results, detailed in Table 6, demonstrate that adding image-EEG data significantly enhances translation performance for both the single convolutional transformer and the multi-view transformer.

Methods	BLEU-N				ROUGE-1		
	N = 1	N = 2	N = 3	N = 4	F	P	R
Masked Token Prediction	0.409	0.242	0.141	0.087	0.325	0.357	0.300
Continuous Masked Token Prediction	0.411	0.243	0.137	0.078	0.319	0.352	0.294
Re-Masked Token Prediction	0.431	0.260	0.157	0.098	0.330	0.361	0.306

Table 5: Ablation study of different pre-training strategies of the EEG signals.

Methods	BLEU-N				ROUGE-1		
	N = 1	N = 2	N = 3	N = 4	F	P	R
Single-View without image-EEG	0.431	0.260	0.157	0.098	0.330	0.361	0.306
Single-View with image-EEG	0.445	0.274	0.175	0.117	0.341	0.383	0.310
Multi-View without image-EEG	0.442	0.277	0.179	0.121	0.335	0.365	0.311
Multi-View with image-EEG	0.452	0.291	0.197	0.141	0.342	0.369	0.320

Table 6: Ablation study of adding image-EEG data into pre-training.

Multi-View Transformer Finally, we compare different training strategies of the multi-view transformer to demonstrate the effectiveness of the multi-view transformer and find the best training strategy. The image-EEG data was not included in this ablation study. Specifically, we compared three training strategies as follows:

- **Only Global Transformer:** Fixing the parameters of all 10 convolutional transformer modules and training only the global transformer for text decoding.
- **Global Transformer + One Convolutional Transformer:** During each training epoch, train one convolutional transformer with the global transformer while fixing the parameters of the remaining nine convolutional transformers.
- **Global Transformer + Three Convolutional Transformers:** During each training epoch, train three convolutional transformers with the global transformer while fixing the parameters of the remaining seven convolutional transformers.

The results in Table 7 demonstrate that activating three convolutional transformers together with the global transformer achieves the best performance.

This suggests further improvement may be attainable by increasing the number of activated convolutional transformers during each training epoch if more GPU resources are available.

Case Study Table 8 shows our case study results. In the first sentence, the baseline model accurately translates “good,” whereas EEG2TEXT, in addition, accurately captures the first half of the sentence with “movie” (synonymous with “film”). Additionally, EEG2TEXT correctly translates the second half of the sentence with “disaster movie” corresponding to “monstrous one” in the original sentence. In the second sentence, EEG2TEXT accurately captured “won Nobel Prize in Chemistry,” while the baseline produced incorrect information, stating “Pulitzer Prize” and the wrong field, “Literature.” In the third sentence, both EEG2TEXT and the baseline correctly identified “book” and “Pulitzer

Prize.” However, EEG2TEXT, in addition, correctly identified the field as “Biography,” while the baseline erroneously outputted “Fictionography.”

In addition, we conducted an interesting case study to show that EEG2TEXT has the ability of zero-shot image-to-text translation. Details can be found in Appendix B.

5 Related Work

Brain Computer Interface The landscape of brain-to-speech and brain-to-text decoding encompasses three principal approaches grounded in the features they capture: motor imagery-based, overt speech-based, and inner speech-based. These methods explore a variety of brain signals, including electroencephalogram (EEG), electrocorticography (ECoG), and functional magnetic resonance imaging (fMRI). Despite these endeavors, existing approaches exhibit limitations concerning vocabulary size, articulation dependence, speed, and device compatibility. Motor imagery-based systems, exemplified by point-and-click [Pandarinath *et al.*, 2017] mechanisms and imaginary handwriting [Willett *et al.*, 2021], showcase high accuracy but modest typing rates. Overt speech-based techniques for decoding or synthesizing speech offer expedited communication rates. However, they require either physical vocal tract movement [Herff *et al.*, 2015; Anumanchipalli *et al.*, 2019; Makin *et al.*, 2020] or mental articulation imagination [Moses *et al.*, 2021; Willett *et al.*, 2023]. This engenders language dependency and pronunciation variations across languages. Another line of research tackles articulation dependency by decoding imagined speech [Nieto *et al.*, 2022] or reading text [Sun *et al.*, 2019; Panachakel and Ramakrishnan, 2021].

Our work follows this line of decoding reading text directly from EEG signals.

EEG-to-Text Decoding Prior investigations into the decoding of EEG-to-text, as documented in the literature [Herff *et al.*, 2015; Sun *et al.*, 2019; Anumanchipalli *et al.*, 2019; Makin *et al.*, 2020; Panachakel and Ramakrishnan, 2021; Moses *et al.*, 2021; Nieto *et al.*, 2022], have

Methods	BLEU-N				ROUGE-1		
	N = 1	N = 2	N = 3	N = 4	F	P	R
Only Global Transformer	0.404	0.238	0.139	0.084	0.303	0.335	0.279
+ One Convolutional Transformer	0.436	0.270	0.168	0.110	0.327	0.363	0.299
+ Three Convolutional Transformers	0.442	0.277	0.179	0.121	0.335	0.365	0.311

Table 7: Ablation study of different training strategies of the multi-view transformer.

(1)	Ground Truth: It's not a particularly good film , but neither is it a monsterous one. Baseline Output: was a a bad good story, but it is it bad bad . one. EEG2TEXT output: 's a a great romantic movie , but it is it the disaster movie one.
(2)	Ground Truth: He won a Nobel Prize in Chemistry in 1928 Baseline Output: was the Pulitzer Prize for Literature in 18. EEG2TEXT Output: won Nobel Prize in Chemistry for 1901
(3)	Ground Truth: The book was awarded the 1957 Pulitzer Prize for Biography . Baseline Output: first is published the Pulitzer Pulitzer Prize for Fictionography. EEG2TEXT Output: book is a Pulitzer Prize for Biography .

Table 8: Case study of the output sentences comparing EEG2TEXT and the baseline method [Wang and Ji, 2021].

demonstrated commendable accuracy when applied to limited and closed vocabularies. Nevertheless, these studies encounter challenges in attaining comparable levels of accuracy when confronted with more extensive and open vocabularies. New investigations have expanded their focus from closed-vocabulary EEG-to-text decoding to encompass open-vocabulary scenarios [Wang and Ji, 2021; Willett *et al.*, 2023; Tang *et al.*, 2023; Duan *et al.*, 2023]. The two research studies most similar to our work are a baseline method [Wang and Ji, 2021] and DeWave [Duan *et al.*, 2023]. The baseline method proposes a framework utilizing transformer and pre-trained BART language models, which establish baseline performance of open-vocabulary EEG-to-text translation. DeWave employs a quantization encoder to derive discrete encoding and aligns it with a pre-trained language model for the open-vocabulary EEG-to-text translation. The limitations of both the baseline method and DeWave lie in their reliance on eye-tracking calibration for word-level EEG feature extraction that introduces error propagation and lacks generalizability to scenarios such as inner speech decoding. EEG2TEXT improves the open-vocabulary EEG-to-text translation performance as well as enhancing the generality by requiring only sentence-level EEG signals as input.

EEG Encoding It is a challenging problem to effectively encode the long and noisy EEG signals to facilitate subsequent decoding tasks. In Conformer [Song *et al.*, 2022], the authors propose a compact convolutional transformer, named EEG Conformer, to encapsulate local and global features in a unified EEG classification framework. Specifically, the convolution module learns the low-level local features throughout the one-dimensional temporal and spatial convolution layers. The self-attention module is straightforwardly connected to extract the global correlation within the local temporal fea-

tures. However, in the case of the Conformer model, the authors trained this model from scratch, whereas EEG2TEXT further incorporated pre-training and multi-view settings to enhance the text translation performance.

EEG Pre-Training Recent work, such as BrainBERT [Wang *et al.*, 2023], BENDR [Kostas *et al.*, 2021] and MAEEG [Chien *et al.*, 2022], has been done on EEG signal pre-training that greatly inspired EEG2TEXT.

BrainBERT converts intracranial recordings to spectrograms and uses spectrograms as input. BrainBERT then masks multiple continuous bands of random frequencies and time intervals from spectrograms and aims to reconstruct the original spectrogram. BENDR uses raw EEG signals as input. After a convolutional layer, the raw EEG signals are converted to embedding features. These embedding features are masked by using masked token prediction [Devlin *et al.*, 2018] and the reconstruction goal is the original embedding features. MAEEG uses raw EEG signals as input and masks the embedding features of the convolutional layer generated with a masked token prediction as BENDR. However, MAEEG’s reconstruction goal is the raw EEG signals. EEG2TEXT directly masks the raw EEG signals with the pre-training objective to reconstruct the raw EEG signals. EEG2TEXT also experimented with various masking strategies and incorporated EEG signals for the pre-training process.

6 Conclusion

In this work, we proposed a novel EEG-to-text decoding model, EEG2TEXT that takes raw EEG signals as input and leverages EEG pre-training and a multi-view transformer to enhance the decoding performance. EEG2TEXT achieved

superior performance for open-vocabulary EEG-to-text decoding. Future work includes expanding the model’s capabilities to EEG signals from diverse multi-modal data.

Ethics Statement

Given our current methodology design, we believe that no significant ethical concerns are likely to arise. We have diligently used openly accessible datasets and models, enhancing transparency and accessibility in our study on EEG signal processing - a task gaining attention in Brain-Computer Interface (BCI) research.

However, it is crucial to note that our architectural framework relies on the pre-trained model, BART, which may make biased decisions influenced by its pre-training data. While our experiments have not shown explicit performance issues due to biases, we must recognize that this observation may be limited to the specific dataset and pre-trained model we used. It is essential to stay vigilant and continue exploring methods to mitigate and correct potential biases that could arise when using pre-trained models.

References

- [Aflalo *et al.*, 2015] Tyson Aflalo, Spencer Kellis, Christian Klaes, Brian Lee, Ying Shi, Kelsie Pejsa, Kathleen Shandfield, Stephanie Hayes-Jackson, Mindy Aisen, Christi Heck, et al. Decoding motor imagery from the posterior parietal cortex of a tetraplegic human. *Science*, 348(6237):906–910, 2015.
- [Anumanchipalli *et al.*, 2019] Gopala K Anumanchipalli, Josh Chartier, and Edward F Chang. Speech synthesis from neural decoding of spoken sentences. *Nature*, 568(7753):493–498, 2019.
- [Bouton *et al.*, 2016] Chad E Bouton, Ammar Shaikhouni, Nicholas V Annetta, Marcia A Bockbrader, David A Friedenberg, Dylan M Nielson, Gaurav Sharma, Per B Sederberg, Bradley C Glenn, W Jerry Mysiw, et al. Restoring cortical control of functional movement in a human with quadriplegia. *Nature*, 533(7602):247–250, 2016.
- [Chien *et al.*, 2022] Hsiang-Yun Sherry Chien, Hanlin Goh, Christopher M. Sandino, and Joseph Y. Cheng. Maeeg: Masked auto-encoder for eeg representation learning, 2022.
- [Cochran *et al.*, 1967] William T Cochran, James W Cooley, David L Favin, Howard D Helms, Reginald A Kaenel, William W Lang, George C Maling, David E Nelson, Charles M Rader, and Peter D Welch. What is the fast fourier transform? *Proceedings of the IEEE*, 55(10):1664–1674, 1967.
- [Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.
- [Duan *et al.*, 2023] Yiqun Duan, Jinzhao Zhou, Zhen Wang, Yu-Kai Wang, and Chin-Teng Lin. Dewave: Discrete eeg waves encoding for brain dynamics to text translation, 2023.
- [Gifford *et al.*, 2022] Alessandro T. Gifford, Kshitij Dwivedi, Gemma Roig, and Radoslaw M. Cichy. A large and rich eeg dataset for modeling human visual object recognition. *NeuroImage*, 264:119754, 2022.
- [Herff *et al.*, 2015] Christian Herff, Dominic Heger, Adriana De Pesters, Dominic Telaar, Peter Brunner, Gerwin Schalk, and Tanja Schultz. Brain-to-text: decoding spoken phrases from phone representations in the brain. *Frontiers in neuroscience*, 9:217, 2015.
- [Hochberg *et al.*, 2012] Leigh R Hochberg, Daniel Bacher, Beata Jarosiewicz, Nicolas Y Masse, John D Simeral, Joern Vogel, Sami Haddadin, Jie Liu, Sydney S Cash, Patrick Van Der Smagt, et al. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485(7398):372–375, 2012.
- [Hollenstein *et al.*, 2018] Nora Hollenstein, Jonathan Rotsztejn, Marius Troendle, Andreas Pedroni, Ce Zhang, and Nicolas Langer. Zuco, a simultaneous eeg and eye-tracking resource for natural sentence reading. *Scientific data*, 5(1):1–13, 2018.
- [Joshi *et al.*, 2019] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. Spanbert: Improving pre-training by representing and predicting spans. *CoRR*, abs/1907.10529, 2019.
- [Kostas *et al.*, 2021] Demetres Kostas, Stephane Arocena-Ouellette, and Frank Rudzicz. Bendlr: using transformers and a contrastive self-supervised learning task to learn from massive amounts of eeg data. *Frontiers in Human Neuroscience*, 15:653659, 2021.
- [Lewis *et al.*, 2019] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.
- [Liu *et al.*, 2019] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.
- [Lorach *et al.*, 2023] Henri Lorach, Andrea Galvez, Valeria Spagnolo, Felix Martel, Serpil Karakas, Nadine Interling, Molywan Vat, Olivier Faivre, Cathal Harte, Salif Komi, et al. Walking naturally after spinal cord injury using a brain-spine interface. *Nature*, pages 1–8, 2023.
- [Makin *et al.*, 2020] Joseph G Makin, David A Moses, and Edward F Chang. Machine translation of cortical activity to text with an encoder-decoder framework. *Nature neuroscience*, 23(4):575–582, 2020.
- [Martin *et al.*, 2018] Stephanie Martin, Iñaki Iturrate, José del R Millán, Robert T Knight, and Brian N Pasley. Decoding inner speech using electrocorticography: Progress and challenges toward a speech prosthesis. *Frontiers in neuroscience*, 12:422, 2018.
- [Moses *et al.*, 2021] David A Moses, Sean L Metzger, Jessie R Liu, Gopala K Anumanchipalli, Joseph G Makin, Pengfei F Sun, Josh Chartier, Maximilian E Dougherty, Patricia M Liu, Gary M Abrams, et al. Neuroprosthesis for decoding speech in a paralyzed person with anarthria. *New England Journal of Medicine*, 385(3):217–227, 2021.

- [Nagel and Spüler, 2018] Sebastian Nagel and Martin Spüler. Modelling the brain response to arbitrary visual stimulation patterns for a flexible high-speed brain-computer interface. *PloS one*, 13(10):e0206107, 2018.
- [Nalborczyk *et al.*, 2020] Ladislas Nalborczyk, Romain Grandchamp, Ernst HW Koster, Marcela Perrone-Bertolotti, and Hélène Lœvenbruck. Can we decode phonetic features in inner speech using surface electromyography? *PloS one*, 15(5):e0233282, 2020.
- [Nieto *et al.*, 2022] Nicolás Nieto, Victoria Peterson, Hugo Leonardo Rufiner, Juan Esteban Kamienkowski, and Ruben Spies. Thinking out loud, an open-access eeg-based bci dataset for inner speech recognition. *Scientific Data*, 9(1):52, 2022.
- [Panachakel and Ramakrishnan, 2021] Jerrin Thomas Panachakel and Angarai Ganesan Ramakrishnan. Decoding covert speech from eeg-a comprehensive review. *Frontiers in Neuroscience*, 15:392, 2021.
- [Pandarinath *et al.*, 2017] Chethan Pandarinath, Paul Nuyujikian, Christine H Blabe, Brittany L Sorice, Jad Saab, Francis R Willett, Leigh R Hochberg, Krishna V Shenoy, and Jaimie M Henderson. High performance communication by people with paralysis using an intracortical brain-computer interface. *Elife*, 6:e18554, 2017.
- [Song *et al.*, 2022] Yonghao Song, Qingqing Zheng, Bingchuan Liu, and Xiaorong Gao. Eeg conformer: Convolutional transformer for eeg decoding and visualization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:710–719, 2022.
- [Sun *et al.*, 2019] Jingyuan Sun, Shaonan Wang, Jiajun Zhang, and Chengqing Zong. Towards sentence-level brain decoding with distributed representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7047–7054, 2019.
- [Tang *et al.*, 2023] Jerry Tang, Amanda LeBel, Shailee Jain, and Alexander G Huth. Semantic reconstruction of continuous language from non-invasive brain recordings. *Nature Neuroscience*, pages 1–9, 2023.
- [Wang and Ji, 2021] Zhenhailong Wang and Heng Ji. Open vocabulary electroencephalography-to-text decoding and zero-shot sentiment classification. *CoRR*, abs/2112.02690, 2021.
- [Wang *et al.*, 2023] Christopher Wang, Vighnesh Subramanian, Adam Uri Yaari, Gabriel Kreiman, Boris Katz, Ignacio Cases, and Andrei Barbu. Brainbert: Self-supervised representation learning for intracranial recordings. *arXiv preprint arXiv:2302.14367*, 2023.
- [Willett *et al.*, 2021] Francis R Willett, Donald T Avansino, Leigh R Hochberg, Jaimie M Henderson, and Krishna V Shenoy. High-performance brain-to-text communication via handwriting. *Nature*, 593(7858):249–254, 2021.
- [Willett *et al.*, 2023] Francis R Willett, Erin M Kunz, Chaofei Fan, Donald T Avansino, Guy H Wilson, Eun Young Choi, Foram Kamdar, Matthew F Glasser, Leigh R Hochberg, Shaul Druckmann, et al. A high-performance speech neuroprosthesis. *Nature*, 620(7976):1031–1036, 2023.

A EEG to Stectrogram

Figure A1 shows a piece of EEG signals and its corresponding spectrogram.

B Zero-Shot Image-to-Text Translation

Figure A2a and A2b show the zero-shot image-to-text translation results. We directly input the EEG signals of image-EEG data into the multi-view transformer model after training, and the output results are image-to-text translation results. The first image contains multiple cars, and the output accurately captures the "car" keyword. The second image contains a fish, and the output captures the "fish" keyword equally accurately.

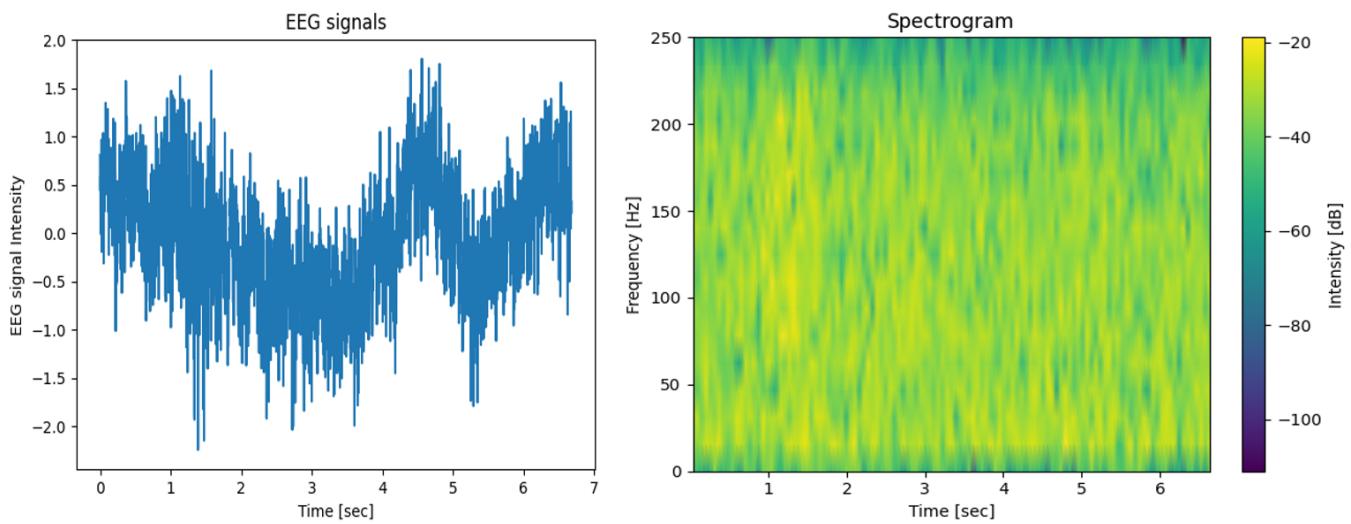


Figure A1: a piece of EEG signals and its corresponding Spectrogram



(a) An image of car. The translation result of EEG2TEXT is: "alog,, car,,,...,,,...,,,"



(b) An image of car. The translation result of EEG2TEXT is: "fish,,... has,,,...,,,...,,,"

Figure A2: Zero-Shot Image-to-Text Translation.

LARGE BRAIN MODEL FOR LEARNING GENERIC REPRESENTATIONS WITH TREMENDOUS EEG DATA IN BCI

Wei-Bang Jiang¹, Li-Ming Zhao^{2*} & Bao-Liang Lu^{12*}

¹Shanghai Jiao Tong University ²Shanghai Emotionhelper Technology Co., Ltd.

935963004@sjtu.edu.cn, liming.zhao@emotionhelper.com, bllu@sjtu.edu.cn

ABSTRACT

The current electroencephalogram (EEG) based deep learning models are typically designed for specific datasets and applications in brain-computer interaction (BCI), limiting the scale of the models and thus diminishing their perceptual capabilities and generalizability. Recently, Large Language Models (LLMs) have achieved unprecedented success in text processing, prompting us to explore the capabilities of Large EEG Models (LEMs). We hope that LEMs can break through the limitations of different task types of EEG datasets, and obtain universal perceptual capabilities of EEG signals through unsupervised pre-training. Then the models can be fine-tuned for different downstream tasks. However, compared to text data, the volume of EEG datasets is generally small and the format varies widely. For example, there can be mismatched numbers of electrodes, unequal length data samples, varied task designs, and low signal-to-noise ratio. To overcome these challenges, we propose a unified foundation model for EEG called Large Brain Model (LaBraM). LaBraM enables cross-dataset learning by segmenting the EEG signals into EEG channel patches. Vector-quantized neural spectrum prediction is used to train a semantically rich neural tokenizer that encodes continuous raw EEG channel patches into compact neural codes. We then pre-train neural Transformers by predicting the original neural codes for the masked EEG channel patches. The LaBraMs were pre-trained on about 2,500 hours of various types of EEG signals from around 20 datasets and validated on multiple different types of downstream tasks. Experiments on abnormal detection, event type classification, emotion recognition, and gait prediction show that our LaBraM outperforms all compared SOTA methods in their respective fields. Our code is available at <https://github.com/935963004/LaBraM>.

1 INTRODUCTION

Electroencephalography (EEG) is a method to record an electrogram of the spontaneous electrical activity of the brain. It is typically non-invasive, with the EEG electrodes placed along the scalp using the international 10–20 system. EEG signals can be formulated as a matrix of real numbers $X \in \mathbb{R}^{C \times T}$, where C is the number of EEG electrodes (channels) that may vary depending on the acquisition equipment used, and T represents the total number of samples, which is related to the collection time and sampling rate. As highly objective physiological signals, EEG has demonstrated remarkable potential in seizure epilepsy classification (Boonyakanont et al., 2020), acute stress detection (Sharma et al., 2022), sleep stage classification (Aboalayon et al., 2016), motor imagery recognition (Amin et al., 2019), abnormal identification (Roy et al., 2019), emotion analysis (Suhaimi et al., 2020), and auditory attention detection (Biesmans et al., 2016).

Numerous deep learning models have been proposed to address the aforementioned tasks in their respective fields. Some works apply convolutional neural networks (CNN) across and within raw EEG channels to encode spatial and temporal features (Lawhern et al., 2018), while others preprocess the data using short-time Fourier transform (STFT) and employ Graph Neural Network (GNN)

*Li-Ming Zhao and Bao-Liang Lu are co-corresponding authors.

on the resulting spectrograms to obtain semantic features of brain area links (Song et al., 2018). Researchers also segment the signal and use a CNN segment encoder with a downstream sequence model such as recurrent neural networks (RNN) to capture temporal dynamics (Xu et al., 2020). These models primarily focus on EEG samples that adhere to specific task formats, mainly because the equipment used to collect EEG differs between datasets, which introduces mismatched channels and variable lengths. Meanwhile, EEG data collection is quite expensive, which makes it challenging to build large EEG datasets specifically designed for a particular task. To prevent overfitting, the parameters of these models need to be regulated, which in turn hampers the model’s ability to learn EEG expressions and limits its generalizability. Consequently, we discovered that current EEG models are typically proprietary and lack the capacity to perform cross-task learning.

Recently, we have been impressed by the capabilities of LLMs (Ouyang et al., 2022; Wei et al., 2022). Specifically, Transformer-based models have demonstrated promising results in natural language processing tasks, which highlights the potential of self-supervised pre-training as a means for harnessing large-scale data. These masked language modeling tasks involve randomly masking some proportion of tokens within a text and then recovering the masked tokens based on the Transformer encoding results of the corrupted text. Motivated by these methods, we propose to apply reconstruction ideas to pre-train neural Transformers. However, it is a daunting task to directly apply LLM-style pre-training to EEG data. The challenges are summarized as follows:

1) Lack of sufficient EEG data. The acquisition of EEG data is significantly challenging compared to natural language and image data. Moreover, the annotation of EEG data usually requires a lot of effort on the part of experts in the corresponding field, thus leading to the fact that only small labeled datasets exist for specific tasks in BCI, where EEG signals are often collected from a small number of participants, typically less than tens of hours in duration. As a result, there is currently no single EEG dataset that is large enough to support the training of LEMs. It remains problems **Q1: how to utilize large-scale unlabeled EEG data?** and **Q2: how much data is needed to train LEMs?**.

2) Diverse configurations of EEG collection. Despite the availability of the international 10-20 system to ensure standardization in EEG testing, users may choose to collect data using EEG caps with different electrode numbers or patch electrodes based on their practical application needs. Thus, how to handle the diverse formats of EEG data in order to match the input units of neural Transformers remains a significant research endeavor.

3) Lack of effective EEG representation learning paradigm. Low signal-to-noise ratio (SNR) and different types of noise are the greatest challenges. Additionally, balancing temporal and spatial characteristics is crucial for effective EEG representation learning. Despite the availability of various deep learning-based EEG representation learning paradigms, such as CNN, RNN, and GNN, for raw EEG data, many researchers still prefer to design artificial EEG features due to these challenges.

In this paper, our objective is to devise a versatile large EEG model that can efficiently handle diverse EEG datasets with varying channels and lengths. By utilizing unsupervised training on a substantial amount of EEG data, we envision the model to possess universal EEG data comprehension capabilities, enabling it to quickly adapt to various EEG downstream tasks. We collected over 2,500 hours of diverse EEG data across various tasks and formats from about 20 datasets. These datasets were primarily obtained from publicly available EEG datasets, as well as our own collected EEG data. Raw EEG signals were first segmented into EEG channel patches to deal with the issues of variant electrodes and time length. Vector-quantized neural spectrum prediction is used to train a semantically rich neural tokenizer to generate neural vocabulary. Specifically, the tokenizer was trained by predicting the Fourier spectrum of the original signal. During pre-training, part of EEG patches are masked while the objective of the neural Transformer is to predict masked tokens from visible patches. We pre-trained three models with varying parameter sizes, ranging from 5.8M to 369M, which are the largest models in BCI ever, and fine-tuned them on four distinct types of downstream tasks encompassing both classification and regression. The contributions of this work are summarized as follows:

- **Large-scale EEG pre-training.** We collected and pre-trained a large-scale neural Transformer model on more than 2,500 hours of diverse EEG data. As far as we know, this is the first time such extensive and varied datasets have been utilized for EEG pre-training.
- **Being compatible with various EEG configurations.** LaBraMs are unified models that are able to handle EEG signals with various channels and time lengths with the assistance of the flexible

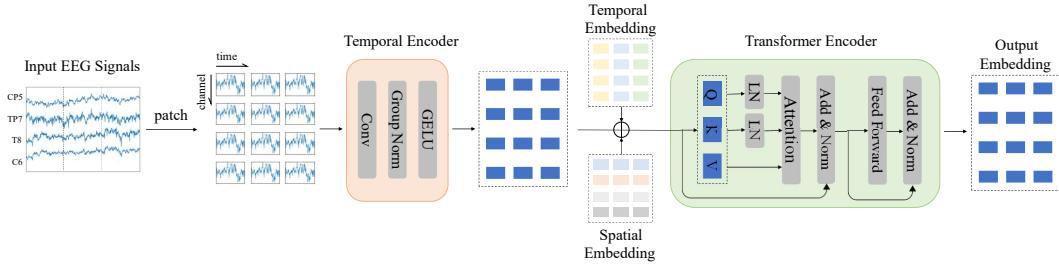


Figure 1: The overall architecture of LaBraM, i.e., neural Transformer. All input EEG signals will first be segmented into EEG patches through a fixed-length time window, and then a temporal encoder will be applied to each patch to extract temporal features. Afterward, temporal and spatial embeddings are added to the patch features to carry temporal and spatial information. At last, the sequence of embeddings is passed into the Transformer encoder by patch-wise attention to obtain the final output.

spatial and temporal embeddings. Hence, one pre-trained LaBraM can adapt to any downstream dataset with different configurations.

- **Effective EEG representation learning.** The utilization of the neural Transformer allows the model to effectively capture both temporal and spatial features of EEG signals with varying channels and lengths, making it suitable for a wide range of downstream tasks in EEG analysis. We further define a neural codebook that offers a compact, versatile, and meaningful representation of EEG signals. We resolve **Q1** by leveraging this codebook to pre-train LaBraM by masked EEG modeling. The empirical performance demonstrates the effectiveness of our proposed method and paves the way for further development in aligning this codebook with natural language.
- **Comprehensive experiments on downstream datasets.** We evaluate our LaBraMs on four representative downstream tasks in BCI, where they surpass all SOTA methods by a large margin. Additionally, we conduct experiments to answer **Q2** by scaling the pre-training data size and conclude the amount of pre-training data required for models of different sizes in Section 3.6.

2 METHOD

In this section, we detail the whole framework of LaBraM. We first formulate the multi-channel EEG signals as $X \in \mathbb{R}^{C \times T}$, where C is the number of EEG electrodes (channels) and T is the total timestamps. The electrode set of X is formulated as $\mathcal{C}_X = \{c_{i_1}, c_{i_2}, \dots, c_{i_C}\}$, where $\mathcal{C}_X \subseteq \mathcal{C} = \{c_1, c_2, \dots, c_{|\mathcal{C}|}\}$ and \mathcal{C} is the universal set of channels in the international 10-20 system.

2.1 MODEL ARCHITECTURE

We introduce the neural Transformer, a general architecture for decoding EEG signals that can deal with any input EEG signals with arbitrary number of channels and time length, as illustrated in Figure 1. The key operation for achieving this is segmenting the EEG signals into patches, inspired by patch embeddings in images (Dosovitskiy et al., 2021). Assume that the timestamp for each sample is t and the stride is s . X can be segmented into $\lfloor \frac{T-t}{s} \rfloor + 1$ samples, and each sample $x \in \mathbb{R}^{C \times t}$. We use a w -length window without overlap to segment each EEG channel into patches, obtaining $x = \{x_{c_{i_j}, k} \in \mathbb{R}^w | j = 1, 2, \dots, C, k = 1, 2, \dots, \lfloor \frac{t}{w} \rfloor\}$. The total number of the patches x is $|x| = C \lfloor \frac{t}{w} \rfloor$.

Temporal Encoder. As EEG is of high resolution in the temporal domain, it is vital to extract temporal features before patch-wise interaction by self-attention. We employ a temporal encoder which consists of several temporal convolution blocks to encode each EEG patch into a patch embedding. The temporal convolution block is composed of a 1-D convolution layer, a group normalization layer (Wu & He, 2018), and a GELU activation function (Hendrycks & Gimpel, 2016). We denote the output patch embeddings from the temporal encoder as

$$\mathbf{e} = \{e_{c_{i_j}, k} \in \mathbb{R}^d | j = 1, 2, \dots, C, k = 1, 2, \dots, \lfloor \frac{t}{w} \rfloor\}, \quad (1)$$

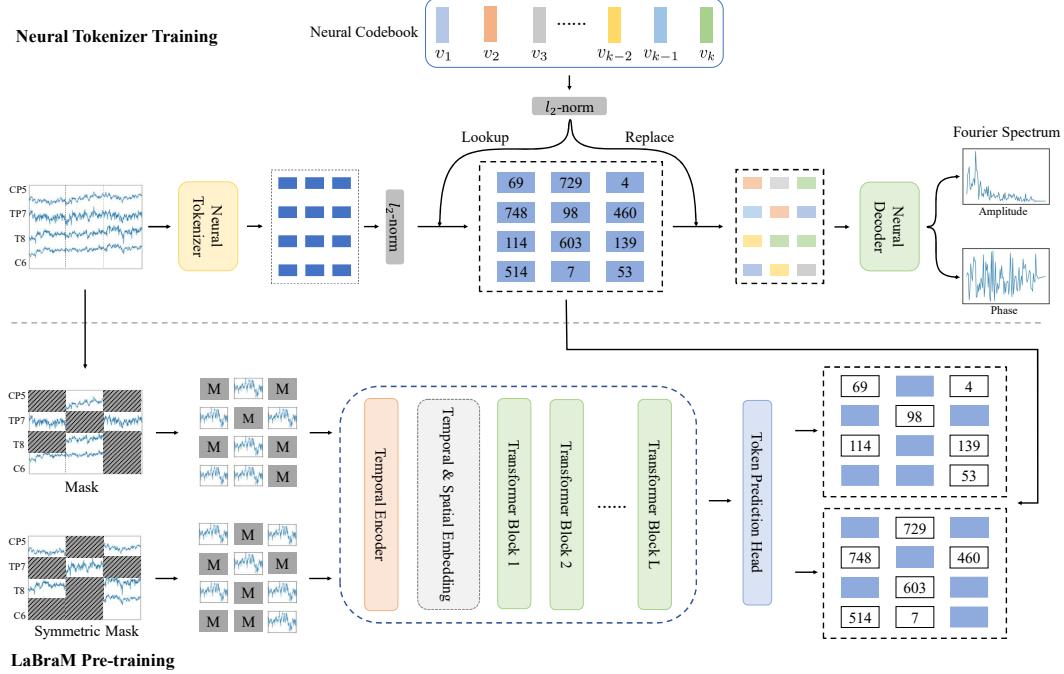


Figure 2: Overview of neural tokenizer training and LaBrA-M pre-training. **Up:** We train a neural tokenizer to discretize EEG signals into discrete neural tokens by reconstructing the Fourier spectrum. **Down:** During pre-training, part of EEG patches are masked while the objective is to predict masked tokens from visible patches.

where d is the dimension of the embeddings.

Temporal & Spatial Embedding. In order to enable the model to be aware of the temporal and spatial information of patch embeddings, we initialize a temporal embedding list $TE = \{te_1, te_2, \dots, te_{tmax}\}$ and a spatial embedding list $SE = \{se_1, se_2, \dots, se_{|C|}\}$, both of which are d -dimension and are set learnable during training. Note that $tmax$ is the hyperparameter determining the maximum number of time patches and $\lfloor \frac{t}{w} \rfloor \leq tmax$. Meanwhile, for each channel c_i , we can find its corresponding spatial embedding se_i in the spatial embedding list SE . Thus, given one arbitrary output embedding $e_{c_i, k}$ in Equation 1 from the temporal encoder, we add the corresponding temporal and spatial embeddings to it:

$$\{e_{c_i, k} + te_k + se_i | j = 1, 2, \dots, C, k = 1, 2, \dots, \lfloor \frac{t}{w} \rfloor\}, \quad (2)$$

where temporal and spatial embeddings act as absolute position encoding.

Transformer Encoder. Finally, the sequence of embeddings will be directly fed into the Transformer encoder (Vaswani et al., 2017). To make the training of Transformer more stable and efficient, we incorporate some modifications (Dehghani et al., 2023). First, we add layer normalization to the queries and keys before the dot-product attention mechanism, which avoids over-large values in attention logits:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{\text{LN}(Q)\text{LN}(K)^T}{\sqrt{d_{head}}}\right)V, \quad (3)$$

where d_{head} is the dimension of one head in the multi-head attention and LN denotes the layerNorm (Ba et al., 2016). Next, we omit the bias term in QKV computations, which accelerates the training without performance degradation. For downstream tasks, we use average pooling on the output embeddings followed by task-specific prediction heads.

2.2 NEURAL TOKENIZER TRAINING

Prior to pre-training LaBraM through masking and prediction, we need to tokenize the EEG into discrete tokens. We propose the vector-quantized neural spectrum prediction, which is trained by predicting the Fourier spectrum, as shown in Figure 2. The key components are the neural tokenizer which encodes EEG samples into patch representations and the neural decoder which decodes the Fourier spectrum from neural embeddings. The idea is basically inspired by VQ-VAE (Van Den Oord et al., 2017) which encodes images into discrete latent representations.

Neural Tokenizer. We define a neural codebook $\mathcal{V} = \{v_i | i = 1, \dots, K\} \in \mathbb{R}^{K \times D}$, where K is the number of the discrete neural embeddings and D is the dimensionality of each embedding. Given an EEG signal sample x , the neural tokenizer whose backbone is just described in Section 2.1 first encode it to patch representations $p = \{p_i | i = 1, \dots, N\}$, where $N = C \lfloor \frac{t}{w} \rfloor$. After that, we utilize a quantizer to quantize all the patch representations into the neural codebook embeddings. The codebook looks up the nearest neighbor of each patch p_i in the neural codebook \mathcal{V} . This procedure can be formulated as

$$z_i = \arg \min_j \|\ell_2(p_i) - \ell_2(v_i)\|_2, \quad (4)$$

where ℓ_2 represents ℓ_2 normalization and z_i is the quantized vector after the quantizer. This is equivalent to finding the closest neural embedding by cosine similarity and such ℓ_2 normalization improves the codebook utilization (Peng et al., 2022).

Fourier Spectrum Prediction. Unlike images that are of high signal-to-noise ratio, EEG signals are of low signal-to-noise ratio and have characteristics of apparent stochasticity, nonstationarity, and nonlinearity nature, which make it hard to reconstruct the original signals well (Moss et al., 2004). In our previous experiments, the loss fails to converge while directly reconstructing raw EEG signals. Instead, the frequency and phase distribution from the Fourier spectrum of EEG signals reveals the underlying neurophysiological activities of the brain (Wu et al., 2022). Therefore, we propose to reconstruct the amplitude and phase from discrete neural tokens for training the neural tokenizer and neural decoder. For an EEG patch $x_{c,k} = [x[1], x[2], \dots, x[w]]$ of channel c and time k in a sample x , we apply the Discrete Fourier Transform (DFT) as follows

$$\tilde{x}_{c,k}^m = \sum_{n=1}^N x[n] \exp\left(-\frac{2\pi j}{N} mn\right), \quad (5)$$

where $m \in [1, N]$ and j is the imaginary unit. We rewrite Equation 5 using Euler's formula as

$$\tilde{x}_{c,k}^m = \sum_{n=1}^N x[n] \cos\left(\frac{2\pi}{N} mn\right) - j x[n] \sin\left(\frac{2\pi}{N} mn\right). \quad (6)$$

Note that $\tilde{x}_{c,k}^m$ indicates the spectrum of the sequence at frequency $\omega_m = \frac{2\pi m}{N}$. Consequently, the amplitude and phase can be calculated as

$$A^m = \sqrt{\operatorname{Re}(\tilde{x}_{c,k}^m)^2 + \operatorname{Im}(\tilde{x}_{c,k}^m)^2}, \quad (7)$$

$$\phi^m = \arctan\left(\frac{\operatorname{Im}(\tilde{x}_{c,k}^m)}{\operatorname{Re}(\tilde{x}_{c,k}^m)}\right), \quad (8)$$

where Re and Im stand for the real and imaginary parts of a complex number. It is worthwhile to mention that we adopt z-score normalization to normalize A^m and ϕ^m within a sample for stable convergence.

After being tokenized by the quantizer, the normalized discrete neural embeddings $\{\ell_2(v_{z_i}) | i = 1, \dots, N\}$ are passed into the neural decoder that comprises several Transformer blocks. The output representations are aggregated by average pooling followed by two specific prediction heads to regress the spectrum amplitude o^A and phase o^ϕ , respectively. The mean squared error (MSE) loss is utilized to guide the prediction. Ultimately, the total loss for training the vector-quantized neural spectrum prediction is defined as

$$\mathcal{L}_T = \sum_{x \in \mathcal{D}} \sum_{i=1}^N \|o_i^A - A_i\|_2^2 + \|o_i^\phi - \phi_i\|_2^2 + \|\mathbf{sg}(\ell_2(p_i)) - \ell_2(v_{z_i})\|_2^2 + \|\ell_2(p_i) - \mathbf{sg}(\ell_2(v_{z_i}))\|_2^2, \quad (9)$$

where \mathcal{D} is all EEG data and sg represents the stop-gradient operation that is defined as an identity at the forward pass and has zero gradients. To make the codebook update more stable, we employ the exponential moving average strategy (Van Den Oord et al., 2017).

2.3 PRE-TRAINING LABRAM

Masked EEG Modeling. To enforce LaBraM learning generic representations with tremendous EEG data, we propose masked EEG modeling. The whole procedure is presented in Figure 2. As formulated in Section 2.1, given an EEG sample x , the temporal encoder first transforms it to patch embeddings $e = \{e_i | i = 1, \dots, N\}$. We randomly generate a mask $\mathcal{M} = \{m_i | i = 1, \dots, N\}$ where $m_i \in \{0, 1\}$ with r proportion of m is 1. After that, we replace the masked patches of x with the learnable mask token $e_M \in \mathbb{R}^d$. The corrupted EEG patches can be denoted as $e^{\mathcal{M}} = \{e_i : m_i = 0 | i = 1, \dots, N\} \cup \{e_M : m_i = 1 | i = 1, \dots, N\}$, which will be added by temporal and spatial embeddings, and then fed into Transformer encoder. We denote the output hidden vectors as $h = \{h_i | i = 1, \dots, N\}$, which are used to predict the corresponding neural tokens through a linear classifier:

$$p(v' | e^{\mathcal{M}}) = \text{softmax}(\text{Linear}(h)). \quad (10)$$

Our objective training loss is

$$\mathcal{L}_{\mathcal{M}} = - \sum_{x \in \mathcal{D}} \sum_{m_i=1} \log p(v_i | e^{\mathcal{M}}). \quad (11)$$

Symmetric Masking. We further propose a symmetric masking strategy to improve training efficiency. We calculate the inverse of the generated mask \mathcal{M} , obtaining $\tilde{\mathcal{M}} = \{\sim m_i | i = 1, \dots, N\}$. Similarly, we use the new mask $\tilde{\mathcal{M}}$ to perform the masked EEG modeling, obtaining the masked EEG prediction loss $\mathcal{L}_{\mathcal{M}}^{sym}$. The motivation is from two aspects: 1) Since we introduce the neural tokenizer, there will be an extra computation overhead, i.e., one forward pass for each EEG sample. Thus, the symmetric masking reuses the same discrete representations, thus improving training efficiency. 2) The symmetric masking provides more masking perspectives in one batch, increasing the data divergency. This simple strategy boosts downstream performance as demonstrated in Appendix I.

Finally, the overall training objective for pre-training LaBraM is

$$\mathcal{L} = \mathcal{L}_{\mathcal{M}} + \mathcal{L}_{\mathcal{M}}^{sym}. \quad (12)$$

3 EXPERIMENTS

3.1 EVALUATION DATASETS

We systematically evaluate our LaBraM on the following downstream datasets:

- **TUAB** (abnormal detection) (Obeid & Picone, 2016): A corpus of EEGs which are 23-channel and sampled at 256 Hz. All data have been annotated as normal or abnormal. There are total 409,455 10-second samples that we use for binary classification to predict normal/abnormal.
- **TUEV** (event type classification) (Obeid & Picone, 2016): This corpus is a subset of TUEG that contains annotations of EEG segments as one of six classes: (1) spike and sharp wave (SPSW), (2) generalized periodic epileptiform discharges (GPED), (3) periodic lateralized epileptiform discharges (PLED), (4) eye movement (EYEM), (5) artifact (ARTF) and (6) background (BCKG). The EEG signals contain 23 channels at 256 Hz and are segmented into 112,491 5-second samples.

More experimental results on other BCI tasks can be found in Appendix F.

3.2 EXPERIMENT SETUP

Model Variants. We devise three different configurations of LaBraM: LaBraM-Base, LaBraM-Large, and LaBraM-Huge. The number of parameters is 5.8M for LaBraM-Base, 46M for LaBraM-Large, and 369M for LaBraM-Huge, respectively, which is increased by enlarging the depth of

the Transformer encoder and hidden sizes. More details of the architecture settings are listed in Appendix C. Unless otherwise specified, the results are from LaBraM-Base in this paper.

The time window w of a patch is set to 200 (1 second). To ensure stable computing resource usage, the number of patches (sequence length) is limited to 256. That means, for example, the time length of EEG with 64 (32) channels is set to 4 (8) seconds. As for the window stride (data stride), it is set to 4 seconds in order to cover all training data as well as boost the training speed.

Pre-training & Fine-tuning. For pre-training LaBraM and the vector-quantized neural spectrum prediction, we collect a total time of over 2,500 hours from public datasets and our self-collected data as described in Appendix D. Note that the four downstream datasets are excluded from the pre-training datasets. For the data splitting of TUAB and TUEV, we strictly follow the same strategy as BIOT (Yang et al., 2023a) to compare all methods fairly. Specifically, as the training and test separation is provided by the datasets, we divide the training patients into training and validation groups by 80% and 20%, respectively. We employ binary cross-entropy (BCE) loss for TUAB (binary classification) and cross-entropy loss for TUEV (multi-class classification), respectively. Our experiments are conducted on eight A800 GPUs by Python 3.11.4 and PyTorch 2.0.1 + CUDA 11.8. The best models are trained based on the training set, selected from the validation set, and finally evaluated on the test set. We report the average and standard deviation values on five different random seeds to obtain comparable results. (see Appendix C for more detailed hyperparameters)

Preprocessing. We only employ very little of the necessary preprocessing. We first filter the EEG signals between 0.1 Hz and 75 Hz to remove low-frequency noise. Then, a notch filter of 50 Hz is applied to avoid power-line interference. Finally, all EEG signals are resampled to 200 Hz. As the range of EEG value is typically between -0.1 mV to 0.1 mV, we normalize it by setting the unit to 0.1 mV to guarantee the value mainly between -1 to 1.

Baselines & Metrics. The baselines are from Yang et al. (2023a), where we choose the best results to compare with. We use the following metrics for comparison: 1) **Balanced Accuracy**: the average of recall on each class, which is utilized for both binary and multi-class classification. 2) **AUC-PR**: area under the precision-recall curve for binary classification. 3) **AUROC**: area under the receiver operating characteristic curve, which is used for binary classification as well. 4) **Cohen’s Kappa**: a measure of agreement between categorical variables X and Y , which is calculated from the observed and expected frequencies on the diagonal of a square contingency table. It is used for multi-class classification. 5) **Weighted F1**: A harmonic mean of the precision and recall, where the relative contribution of precision and recall to the F1 score are equal. We use it to evaluate multi-class classification. We set AUROC as the monitor score for binary classification and Cohen’s Kappa as the monitor score for multi-class classification.

3.3 PRE-TRAINING VISUALIZATION

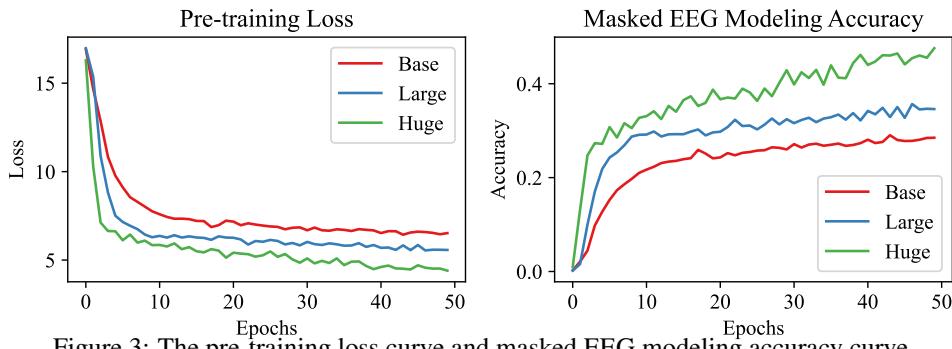


Figure 3: The pre-training loss curve and masked EEG modeling accuracy curve.

Figure 3 compares the convergence curves of the total pre-training loss and masked EEG modeling accuracy between the base, large, and huge models. We observe that a larger model with more parameters can converge to a smaller loss and higher accuracy. Notably, the loss of the huge model seems to have an obvious downward trend while the accuracy tends to increase if we train it longer. This observation suggests scaling up the model size has the potential to obtain better performance.

3.4 COMPARISON WITH STATE-OF-THE-ART

Table 1 and Table 2 present the results of state-of-the-art baselines as well as LaBraM from TUAB and TUEV. The results demonstrate that our LaBraM-Base model outperformed all baselines on various evaluation metrics for both tasks. Particularly in the more challenging multi-class classification task of TUEV, our model achieved a significant improvement in performance. In our own model, we observed that as the number of model parameters increased, the LaBraM-Huge model performed the best, followed by the LaBraM-Large model and then the LaBraM-Base model. We attribute this good performance to the increase in pre-training data volume and model parameters. We believe that with sufficient data volume, large-scale EEG models can learn more generalizable EEG patterns, leading to improved performance on a wide range of downstream tasks in EEG analysis.

Table 1: The results of different methods on TUAB.

Methods	Model Size	Balanced Accuracy	AUC-PR	AUROC
SPaRCNet (Jing et al., 2023)	0.79M	0.7896±0.0018	0.8414±0.0018	0.8676±0.0012
ContraWR (Yang et al., 2023b)	1.6M	0.7746±0.0041	0.8421±0.0104	0.8456±0.0074
CNN-Transformer (Peh et al., 2022)	3.2M	0.7777±0.0022	0.8433±0.0039	0.8461±0.0013
FFCL (Li et al., 2022)	2.4M	0.7848±0.0038	0.8448±0.0065	0.8569±0.0051
ST-Transformer (Song et al., 2021)	3.5M	0.7966±0.0023	0.8521±0.0026	0.8707±0.0019
BIOT (Yang et al., 2023a)	3.2M	0.7959±0.0057	0.8792±0.0023	0.8815±0.0043
LaBraM-Base	5.8M	0.8140±0.0019	0.8965±0.0016	0.9022±0.0009
LaBraM-Large	46M	0.8226±0.0015	0.9130±0.0005	0.9127±0.0005
LaBraM-Huge	369M	0.8258±0.0011	0.9204±0.0011	0.9162±0.0016

Table 2: The results of different methods on TUEV.

Methods	Model Size	Balanced Accuracy	Cohen's Kappa	Weighted F1
SPaRCNet (Jing et al., 2023)	0.79M	0.4161±0.0262	0.4233±0.0181	0.7024±0.0104
ContraWR (Yang et al., 2023b)	1.6M	0.4384±0.0349	0.3912±0.0237	0.6893±0.0136
CNN-Transformer (Peh et al., 2022)	3.2M	0.4087±0.0161	0.3815±0.0134	0.6854±0.0293
FFCL (Li et al., 2022)	2.4M	0.3979±0.0104	0.3732±0.0188	0.6783±0.0120
ST-Transformer (Song et al., 2021)	3.5M	0.3984±0.0228	0.3765±0.0306	0.6823±0.0190
BIOT (Yang et al., 2023a)	3.2M	0.5281±0.0225	0.5273±0.0249	0.7492±0.0082
LaBraM-Base	5.8M	0.6409±0.0065	0.6637±0.0093	0.8312±0.0052
LaBraM-Large	46M	0.6581±0.0156	0.6622±0.0136	0.8315±0.0040
LaBraM-Huge	369M	0.6616±0.0170	0.6745±0.0195	0.8329±0.0086

3.5 PRE-TRAINING WITH/WITHOUT DOWNSTREAM DATASETS

During the pre-training process, we hope that the model can learn general EEG representations that are not specific to any particular task. Although no label data is used during the pre-training process,

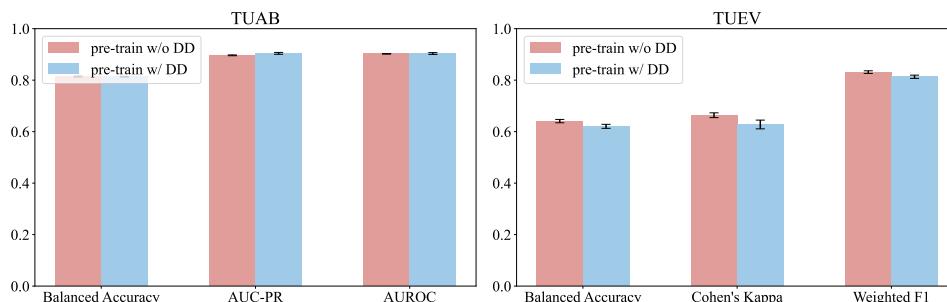


Figure 4: A comparison of the model’s performance on the TUAB and TUEV datasets when incorporating themselves into the pre-training process or not.

to eliminate the influence of the pretraining data on downstream tasks, we compared the results with or without incorporating the downstream task dataset into the pre-training process or not. It is noted that the recordings of TUAB and TUEV are disjoint from recordings of pre-training datasets. As Figure 4 illustrates, the performance of the model on the downstream task was not significantly affected by whether or not to incorporate the downstream task datasets into the model’s pre-training process. This demonstrates that our model has the capability to learn universal EEG representations, and provides guidance for the collection of more EEG data in the future. In other words, we do not need to expend a significant amount of effort on labeling EEG data during the pre-training process.

3.6 SCALING DATA SIZE

Although we have collected approximately 2,500 hours of EEG data, it is still relatively small compared to the sample size in natural language processing and image processing. We answer **Q2** about the demand for data size to train LaBraMs with different sizes by scaling the pre-training data size. As illustrated in Figure 5, the performance of the Base model with 500 hours of training exceeds that of the 2500-hour model on TUAB, while approaching over 90% of the 2500-hour performance on TUEV. For the Large model, performance generally improves with increased data volume, though the growth rate slows after 1000 hours. In contrast, the Huge model exhibits a noticeable upward trend in performance as data size continues to expand. Therefore, we believe that with further expansion of the dataset, our model can achieve better performance. The question of how much EEG data is required for pre-training a large EEG model is undoubtedly an important issue worth exploring in this field. Nevertheless, 2,500 hours is not the answer to this question at least. Our observation basically follows the scaling law (Kaplan et al., 2020), from which we deduce that the Huge model would continue to perform better with the data size on the order of at least ten thousand hours.

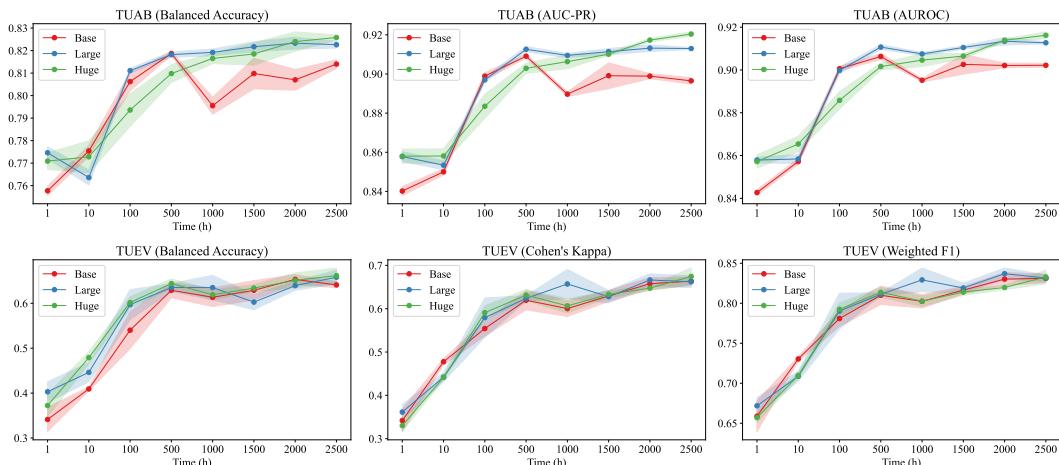


Figure 5: A comparison of the performance of the Base model, Large model, and Huge model on the TUAB and TUEV datasets as the pre-training data increases.

4 CONCLUSION

This paper proposes a Large Brain Model (LaBraM) that learns universal embeddings through unsupervised pre-training on over 2,500 hours of diverse EEG data. The LaBraM is capable of handling diverse EEG datasets due to the segmentation of raw EEG signals into channel patches and the use of vector-quantized neural spectrum prediction to generate a rich semantic tokenizer during pre-training. Additionally, the neural Transformer architecture enables effective representation learning of both temporal and spatial features of EEG signals, making it suitable for a wide range of downstream tasks in EEG analysis. The LaBraM was validated on multiple downstream tasks, including abnormal detection, event type classification, emotion recognition, and gait prediction. Our experiments show that the LaBraM outperforms all SOTA methods in their respective fields. In the end, we hope our work can have implications for future developments in EEG-based deep learning models with improved perceptual capabilities and generalizability.

ACKNOWLEDGMENTS

This work was supported in part by grants from STI 2030-Major Projects+2022ZD0208500, Shanghai Municipal Science and Technology Major Project (Grant No. 2021SHZD ZX), Medical-Engineering Interdisciplinary Research Foundation of Shanghai Jiao Tong University “Jiao Tong Star” Program (YG2023ZD25), and GuangCi Professorship Program of RuiJin Hospital Shanghai Jiao Tong University School of Medicine.

REFERENCES

- Khald Ali I Aboalayon, Miad Faezipour, Wafaa S Almuhammadi, and Saeid Moslehpoor. Sleep stage classification using EEG signal analysis: a comprehensive survey and new investigation. *Entropy*, 18(9):272, 2016.
- Syed Umar Amin, Mansour Alsulaiman, Ghulam Muhammad, Mohamed Amine Mekhtiche, and M Shamim Hossain. Deep Learning for EEG motor imagery classification based on multi-layer CNNs feature fusion. *Future Generation Computer Systems*, 101:542–554, 2019.
- Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.
- Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in Neural Information Processing Systems*, 33:12449–12460, 2020.
- Hubert Banville, Omar Chehab, Aapo Hyvärinen, Denis-Alexander Engemann, and Alexandre Gramfort. Uncovering the structure of clinical EEG signals with self-supervised learning. *Journal of Neural Engineering*, 18(4):046020, 2021.
- Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEit: BERT pre-training of image transformers. In *International Conference on Learning Representations*, 2022. URL <https://openreview.net/forum?id=p-BhZSz59o4>.
- Wouter Biesmans, Neetha Das, Tom Francart, and Alexander Bertrand. Auditory-inspired speech envelope extraction methods for improved EEG-based auditory attention detection in a cocktail party scenario. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(5):402–412, 2016.
- Benjamin Blankertz, Guido Dornhege, Matthias Krauledat, Klaus-Robert Müller, and Gabriel Curio. The non-invasive berlin brain–computer interface: fast acquisition of effective performance in untrained subjects. *NeuroImage*, 37(2):539–550, 2007.
- Poomipat Boonyakanont, Apiwat Lek-Uthai, Krisnachai Chomtho, and Jitkomut Songsiri. A review of feature extraction and performance evaluation in epileptic seizure detection using EEG. *Biomedical Signal Processing and Control*, 57:101702, 2020.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.
- G Buckwalter, S Chhin, S Rahman, I Obeid, and J Picone. Recent advances in the TUH EEG corpus: improving the interrater agreement for artifacts and epileptiform events. In *2021 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pp. 1–3. IEEE, 2021.
- Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, pp. 1691–1703. PMLR, 2020.
- Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, Rodolphe Jenatton, Lucas Beyer, Michael Tschannen, Anurag Arnab, Xiao Wang, Carlos Riquelme Ruiz, Matthias Minderer, Joan Puigcerver, Utku Evci, Manoj Kumar, Sjoerd Van

- Steenkiste, Gamaleldin Fathy Elsayed, Aravindh Mahendran, Fisher Yu, Avital Oliver, Fantine Huot, Jasmijn Bastings, Mark Collier, Alexey A. Gritsenko, Vighnesh Birodkar, Cristina Nader Vasconcelos, Yi Tay, Thomas Mensink, Alexander Kolesnikov, Filip Pavetic, Dustin Tran, Thomas Kipf, Mario Lucic, Xiaohua Zhai, Daniel Keysers, Jeremiah J. Harmsen, and Neil Houlsby. Scaling vision transformers to 22 billion parameters. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 7480–7512. PMLR, 23–29 Jul 2023. URL <https://proceedings.mlr.press/v202/dehghani23a.html>.
- Paolo Detti, Giampaolo Vatti, and Garazi Zabalo Manrique de Lara. Eeg synchronization analysis for seizure prediction: A study on data of noninvasive recordings. *Processes*, 8(7):846, 2020.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=YicbFdNTTy>.
- Xiao Gu, Jinpei Han, Guang-Zhong Yang, and Benny Lo. Generalizable Movement Intention Recognition with Multiple Heterogeneous EEG Datasets. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9858–9864, 2023. doi: 10.1109/ICRA48891.2023.10160462.
- Jinpei Han, Xiaoxi Wei, and A Aldo Faisal. EEG Decoding for Datasets with Heterogenous Electrode Configurations using Transfer Learning Graph Neural Networks. *arXiv preprint arXiv:2306.13109*, 2023.
- Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.
- Yongtian He, Trieu Phat Luu, Kevin Nathan, Sho Nakagome, and Jose L Contreras-Vidal. A mobile brain-body imaging dataset recorded during treadmill walking with a brain-computer interface. *Scientific Data*, 5(1):1–10, 2018.
- Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.
- Wei-Bang Jiang, Li-Ming Zhao, Ping Guo, and Bao-Liang Lu. Discriminating Surprise and Anger from EEG and Eye Movements with a Graph Network. In *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 1353–1357, 2021. doi: 10.1109/BIBM52615.2021.9669637.
- Wei-Bang Jiang, Xuan-Hao Liu, Wei-Long Zheng, and Bao-Liang Lu. Multimodal Adaptive Emotion Transformer with Flexible Modality Inputs on A Novel Dataset with Continuous Labels. In *Proceedings of the 31st ACM International Conference on Multimedia*, MM ’23, pp. 5975–5984, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701085. doi: 10.1145/3581783.3613797. URL <https://doi.org/10.1145/3581783.3613797>.
- Jin Jing, Wendong Ge, Shenda Hong, Marta Bento Fernandes, Zhen Lin, Chaoqi Yang, Sungtae An, Aaron F Struck, Aline Herlopian, Ioannis Karakis, et al. Development of expert-level classification of seizures and rhythmic and periodic patterns during eeg interpretation. *Neurology*, 100(17):e1750–e1762, 2023.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

- Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.
- Louis Korczowski, Martine Cederhout, Anton Andreev, Grégoire Cattan, Pedro Luiz Coelho Rodrigues, Violette Gautheret, and Marco Congedo. Brain Invaders calibration-less P300-based BCI with modulation of flash duration Dataset (bi2015a). Research report, GIPSA-lab, July 2019. URL <https://hal.science/hal-02172347>.
- Demetres Kostas, Stephane Aroca-Ouellette, and Frank Rudzicz. BENDR: using transformers and a contrastive self-supervised learning task to learn from massive amounts of EEG data. *Frontiers in Human Neuroscience*, 15:653659, 2021.
- Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces. *Journal of Neural Engineering*, 15(5):056013, 2018.
- Hongli Li, Man Ding, Ronghua Zhang, and Chunbo Xiu. Motor imagery EEG classification algorithm based on CNN-LSTM feature fusion network. *Biomedical Signal Processing and Control*, 72:103342, 2022.
- Rui Li, Le-Dian Liu, and Bao-Liang Lu. Discrimination of Decision Confidence Levels from EEG Signals. In *2021 10th International IEEE/EMBS Conference on Neural Engineering (NER)*, pp. 946–949, 2021. doi: 10.1109/NER49283.2021.9441086.
- Rui Liu, Yuanyuan Chen, Anran Li, Yi Ding, Han Yu, and Cuntai Guan. Aggregating intrinsic information to enhance BCI performance through federated learning. *Neural Networks*, pp. 106100, 2024.
- Wei Liu, Jie-Lin Qiu, Wei-Long Zheng, and Bao-Liang Lu. Comparing Recognition Performance and Robustness of Multimodal Deep Learning Models for Multimodal Emotion Recognition. *IEEE Transactions on Cognitive and Developmental Systems*, 2021.
- Wei Liu, Wei-Long Zheng, Ziyi Li, Si-Yuan Wu, Lu Gan, and Bao-Liang Lu. Identifying similarities and differences in emotion recognition with EEG and eye movements among Chinese, German, and French People. *Journal of Neural Engineering*, 19(2):026012, 2022.
- Matthew D Luciw, Ewa Jarocka, and Benoni B Edin. Multi-channel EEG recordings during 3,936 grasp and lift trials with varying weight and friction. *Scientific Data*, 1(1):1–11, 2014.
- Shuai Luo, Yu-Ting Lan, Dan Peng, Ziyi Li, Wei-Long Zheng, and Bao-Liang Lu. Multimodal emotion recognition in response to oil paintings. In *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 4167–4170, 2022. doi: 10.1109/EMBC48229.2022.9871630.
- Perrin Margaux, Maby Emmanuel, Daligault Sébastien, Bertrand Olivier, and Mattout Jérémie. Objective and subjective evaluation of online error correction during p300-based spelling. *Advances in Human-Computer Interaction*, 2012:4–4, 2012.
- Frank Moss, Lawrence M Ward, and Walter G Sannita. Stochastic resonance and sensory information processing: a tutorial and review of application. *Clinical Neurophysiology*, 115(2):267–281, 2004.
- Iyad Obeid and Joseph Picone. The temple university hospital EEG data corpus. *Frontiers in Neuroscience*, 10:196, 2016.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022.
- Wei Yan Peh, Yuanyuan Yao, and Justin Dauwels. Transformer convolutional neural networks for automated artifact detection in scalp EEG. In *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 3599–3602. IEEE, 2022.

- Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*, 2022.
- Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- Subhrajit Roy, Isabell Kiral-Kornek, and Stefan Harrer. ChronoNet: A deep recurrent neural network for abnormal EEG identification. In *Artificial Intelligence in Medicine: 17th Conference on Artificial Intelligence in Medicine, AIME 2019, Poznan, Poland, June 26–29, 2019, Proceedings 17*, pp. 47–56. Springer, 2019.
- Arman Savran, Koray Ciftci, Guillaume Chanel, Javier Cruz_Mota, Luong Hong Viet, Bülent Sankur, Lale Akarun, Alice Caplier, and Michele Rombaut. Emotion detection in the loop from brain signals and facial images. In *eINTERFACE'06-SIMILAR NoE Summer Workshop on Multimodal Interfaces*, 2006.
- Gerwin Schalk, Dennis J McFarland, Thilo Hinterberger, Niels Birbaumer, and Jonathan R Wolpaw. BCI2000: a general-purpose brain-computer interface (BCI) system. *IEEE Transactions on Biomedical Engineering*, 51(6):1034–1043, 2004.
- Vinit Shah, Eva Von Weltin, Silvia Lopez, James Riley McHugh, Lillian Veloso, Meysam Golmohammadi, Iyad Obeid, and Joseph Picone. The temple university hospital seizure detection corpus. *Frontiers in Neuroinformatics*, 12:83, 2018.
- Lakhan Dev Sharma, Vijay Kumar Bohat, Maria Habib, Al-Zoubi Ala’M, Hossam Faris, and Ibrahim Aljarah. Evolutionary inspired approach for mental stress detection using EEG signal. *Expert Systems with Applications*, 197:116634, 2022.
- Tengfei Song, Wenming Zheng, Peng Song, and Zhen Cui. EEG emotion recognition using dynamical graph convolutional neural networks. *IEEE Transactions on Affective Computing*, 11(3): 532–541, 2018.
- Yonghao Song, Xueyu Jia, Lie Yang, and Longhan Xie. Transformer-based spatial-temporal feature learning for EEG decoding. *arXiv preprint arXiv:2106.11170*, 2021.
- Nazmi Sofian Suhaimi, James Mountstephens, Jason Teo, et al. EEG-based emotion recognition: A state-of-the-art review of current trends and opportunities. *Computational Intelligence and Neuroscience*, 2020, 2020.
- Le-Yan Tao and Bao-Liang Lu. Emotion Recognition under Sleep Deprivation Using a Multimodal Residual LSTM Network. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2020. doi: 10.1109/IJCNN48605.2020.9206957.
- Mastaneh Torkamani-Azar, Sumeyra Demir Kanik, Serap Aydin, and Mujdat Cetin. Prediction of reaction time and vigilance variability from spatio-spectral features of resting-state EEG in a long sustained attention task. *IEEE Journal of Biomedical and Health Informatics*, 24(9):2550–2558, 2020.
- Logan Trujillo. Raw EEG Data. 2020. doi: 10.18738/T8/SS2NHB. URL <https://doi.org/10.18738/T8/SS2NHB>.
- Logan T Trujillo, Candice T Stanfield, and Ruben D Vela. The effect of electroencephalogram (EEG) reference choice on information-theoretic measures of the complexity and integration of EEG signals. *Frontiers in Neuroscience*, 11:425, 2017.
- Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in Neural Information Processing Systems*, 30, 2017.

- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fb053c1c4a845aa-Paper.pdf.
- L Veloso, J McHugh, E von Weltin, S Lopez, I Obeid, and J Picone. Big data resources for EEGs: Enabling deep learning research. In *2017 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pp. 1–3. IEEE, 2017.
- Eva von Weltin, Tameem Ahsan, Vinit Shah, Dawer Jamshed, Meysam Golmohammadi, Iyad Obeid, and Joseph Picone. Electroencephalographic slowing: A primary source of error in automatic seizure detection. In *2017 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pp. 1–5. IEEE, 2017.
- Christopher Wang, Vighnesh Subramaniam, Adam Uri Yaari, Gabriel Kreiman, Boris Katz, Ignacio Cases, and Andrei Barbu. BrainBERT: Self-supervised representation learning for intracranial recordings. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=xmcYx_reUn6.
- Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.
- Di Wu, Siyuan Li, Jie Yang, and Mohamad Sawan. neuro2vec: Masked fourier spectrum prediction for neurophysiological representation learning. *arXiv preprint arXiv:2204.12440*, 2022.
- Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.
- Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9653–9663, 2022.
- Gaowei Xu, Tianhe Ren, Yu Chen, and Wenliang Che. A one-dimensional cnn-lstm model for epileptic seizure recognition using eeg signal analysis. *Frontiers in Neuroscience*, 14:578126, 2020.
- Chaoqi Yang, M Brandon Westover, and Jimeng Sun. BIOT: Biosignal transformer for cross-data learning in the wild. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a. URL <https://openreview.net/forum?id=c2LZyTyddi>.
- Chaoqi Yang, Cao Xiao, M Brandon Westover, Jimeng Sun, et al. Self-Supervised Electroencephalogram Representation Learning for Automatic Sleep Staging: Model Development and Evaluation Study. *JMIR AI*, 2(1):e46769, 2023b.
- Ke Yi, Yansen Wang, Kan Ren, and Dongsheng Li. Learning Topology-Agnostic EEG Representations with Geometry-Aware Modeling. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=hiOUySN0ub>.
- W. Zheng, W. Liu, Y. Lu, B. Lu, and A. Cichocki. Emotionmeter: A multimodal framework for recognizing human emotions. *IEEE Transactions on Cybernetics*, pp. 1–13, 2018. ISSN 2168-2267. doi: 10.1109/TCYB.2018.2797176.
- Wei-Long Zheng and Bao-Liang Lu. Investigating critical frequency bands and channels for EEG-based emotion recognition with deep neural networks. *IEEE Transactions on Autonomous Mental Development*, 7(3):162–175, 2015. doi: 10.1109/TAMD.2015.2431497.

A RELATED WORK

Self-supervised Pre-training. In recent years, self-supervised pre-training has made significant progress in natural language processing and computer vision. BERT (Devlin et al., 2018) innovatively proposed the idea of masking part of the input sentences and then reconstructing them. The GPT series (Radford et al., 2018; 2019; Brown et al., 2020) proposed to pre-train large language models by a large corpus of data in an autoregressive way. Both studies improved the fine-tuning performance significantly in various downstream tasks. In computer vision, iGPT (Chen et al., 2020) firstly brought the idea from GPT to pre-train a vision model. BEiT (Bao et al., 2022) pioneerly trained a vision tokenizer and leveraged BERT-like pre-training for training a vision Transformer. MAE (He et al., 2022) and SimMIM (Xie et al., 2022) practiced masked image modeling by simply reconstructing the raw pixels and achieved appreciable improvement.

Learning with Heterogeneous Datasets. MMM introduced a pre-training framework built on the unified topology and obtained topology-agnostic representations (Yi et al., 2023). Han et al. (2023) combined graph neural networks and transfer learning for non-invasive motor imagery EEG decoding with heterogeneous electrode configurations. Gu et al. (2023) developed two networks to learn from the shared and the complete channels across datasets, achieving coherent performance boosts. Liu et al. (2024) proposed a hierarchical personalized Federated Learning EEG decoding framework, enabling datasets with disparate data formats to collaborate in the model training process.

Self-supervised Learning in BCI. Although self-supervised pre-training has achieved great success, its potential in BCI is far from being explored. BENDR (Kostas et al., 2021) adapted Wav2vec 2.0 (Baevski et al., 2020), which uses contrastive learning to learn compressed representations of raw EEG signals. Banville *et al.* investigated temporal context prediction as well as contrastive predictive coding on two clinically relevant problems (Banville et al., 2021). ContraWR (Yang et al., 2023b), Contrast with the World Representation, used global statistics to distinguish signals associated with different sleep stages. BrainBERT (Wang et al., 2023) masks random parts of the stereo-electroencephalographic (SEEG) spectrogram and produce original embeddings with 43.6 hours of data. However, all existing studies either concentrate on specific BCI tasks or only employ small-size datasets and models, leaving room for exploring large-scale EEG data to train large EEG models through self-supervision.

B LABRAM PRE-TRAINING ANALYSIS

The pre-training of LaBraM can be interpreted as the training of a variational autoencoder (Kingma & Welling, 2014; Bao et al., 2022). We denote the original EEG sample as x , the corrupted EEG by masking as x^M , and its Fourier spectrum (amplitude and phase) as \tilde{x} . The focus is on the evidence lower bound (ELBO) of the log-likelihood $p(\tilde{x}|x^M)$, which involves recovering the Fourier spectrum of the original EEG signals from the masked perspective:

$$\sum_{(x_i, x_i^M, \tilde{x}_i) \in \mathcal{D}} \log p(\tilde{x}_i | x_i^M) \geq \sum_{(x_i, x_i^M, \tilde{x}_i) \in \mathcal{D}} (\mathbb{E}_{z_i \sim q_\phi(z|x_i)} (\log p_\psi(\tilde{x}_i | z_i) - D_{KL}(q_\phi(z|x_i), p_\theta(z|x_i^M))), \quad (13)$$

where $q_\phi(z|x)$ represents the neural tokenizer that encodes the EEG sample into discrete neural tokens, $p_\psi(\tilde{x}|z)$ denotes the neural decoder predicting the Fourier spectrum from given neural tokens, and $p_\theta(z|x^M)$ is the LaBraM pre-training for masked EEG modeling, where the LaBraM encoder reconstructs neural tokens from the corrupted EEG input.

The whole framework is optimized through a two-stage procedure as (Van Den Oord et al., 2017). For the first stage, we train the neural tokenizer as a discrete variational autoencoder by minimizing the reconstruction loss $-\mathbb{E}_{z_i \sim q_\phi(z|x_i)} (\log p_\psi(\tilde{x}_i | z_i))$ with a uniform prior. For the second stage, we set q_ϕ as well as p_ψ fixed and learn the prior p_θ by minimizing the loss D_{KL} . For simplicity, $q_\phi(z|x_i)$ is defined as a one-point distribution with the most likely neural tokens $\hat{z}_i = \arg \max_z q_\phi(z|x_i)$. Consequently, we can rewrite Equation 13 as

$$\sum_{(x_i, x_i^M, \tilde{x}_i) \in \mathcal{D}} (\mathbb{E}_{z_i \sim q_\phi(z|x_i)} (\log p_\psi(\tilde{x}_i | z_i) + \log p_\theta(\hat{z}_i | x_i^M))), \quad (14)$$

where the first term is the objective for vector-quantized neural spectrum prediction and the second term is the objective for LaBraM pre-training.

C HYPERPARAMETER SETTINGS

Table 3: Hyperparameters for vector-quantized neural spectrum prediction training.

Hyperparameters		Values
Temporal Encoder	Input channels	{1,8,8}
	Output channels	{8,8,8}
	Kernel size	{15,3,3}
	Stride	{8,1,1}
	Padding	{7,1,1}
Transformer encoder layers	12	
Transformer decoder layers	3	
Hidden size	200	
MLP size	800	
Attention head number	10	
Codebook size	8192×64	
Batch size	1024	
Peak learning rate	5e-5	
Minimal learning rate	1e-5	
Learning rate scheduler	Cosine	
Optimizer	AdamW	
Adam β	(0.9,0.99)	
Weight decay	1e-4	
Total epochs	100	
Warmup epochs	10	
Data stride	200	

Table 4: Hyperparameters for masked EEG pre-training.

Hyperparameters		LaBraM-Base	LaBraM-Large	LaBraM-Huge
Temporal Encoder	Input channels	{1,8,8}	{1,16,16}	{1,32,32}
	Output channels	{8,8,8}	{16,16,16}	{32,32,32}
	Kernel size		{15,3,3}	
	Stride		{8,1,1}	
	Padding		{7,1,1}	
Transformer encoder layers	12	24	48	
Hidden size	200	400	800	
MLP size	800	1600	3200	
Attention head number	10	16	16	
Batch size		512		
Peak learning rate		5e-4		
Minimal learning rate		1e-5		
Learning rate scheduler		Cosine		
Optimizer		AdamW		
Adam β		(0.9,0.98)		
Weight decay		0.05		
Total epochs		50		
Warmup epochs		5		
Data stride		800		
Gradient clipping		3		
Layer scale init	0.1	1e-5	1e-6	
EMA weight		0.996		
Mask ratio		0.5		

Table 5: Hyperparameters for downstream fine-tuning.

Hyperparameters	Values
Batch size	512
Peak learning rate	5e-4
Minimal learning rate	1e-6
Learning rate scheduler	Cosine
Optimizer	AdamW
Adam β	(0.9,0.999)
Weight decay	0.05
Total epochs	50 (B) 30 (L/H)
Warmup epochs	5 (B) 3 (L/H)
Drop path	0.1 (B/L) 0.2 (H)
Layer-wise learning rate decay	0.65 (B) 0.8 (L/H)
Label smoothing (multi-class classification)	0.1

D PRE-TRAINING DATASET DESCRIPTION

We describe the datasets we use for training LaBrA-M here.

Training datasets (for both vector-quantized neural spectrum prediction training and LaBrA-M pre-training, the total time is 2534.78 hours):

- **BCI Competition IV-1** (Blankertz et al., 2007): A motor imagery dataset containing 59 EEG channels at 1000Hz sampling rate for 2 classes of left hand, right hand, foot (+ idle state) for 7 subjects. The recording was made using BrainAmp MR plus amplifiers and an Ag/AgCl electrode cap. (total time: 8.21 hours)
- **Emobrain** (Savran et al., 2006): A multimodal emotion dataset where EEG (64 channels, 1024 Hz) and fNIRS, are recorded by the Biosemi Active 2 acquisition system, including 16 subjects. The emotions were elicited through a selected subset of IAPS dataset. (total time: 4.94 hours)
- **Grasp and Lift EEG Challenge** (Luciw et al., 2014): A dataset containing EEG recordings (32 channels, 500 Hz) of 12 subjects performing grasp-and-lift (GAL) trials. The EEG cap was used in conjunction with a BrainAmp EEG signal amplifier. (total time: 11.72 hours)
- **Inria BCI Challenge** (Margaux et al., 2012): A P300-based spelling dataset including 26 subjects with EEG records (56 channels, 600 Hz) by Ag/AgCl EEG sensors (VSM-CTF compatible system). (total time: 29.98 hours)
- **EEG Motor Movement/Imagery Dataset** (Schalk et al., 2004): A motor imagery dataset consisting of 109 volunteers performing 2 baseline tasks (eye-open and eye-closed), motor movement, and motor imagery (both fists or both feet) with EEG records (64 channels, 160 Hz) using the BCI2000 system. (total time: 47.3 hours)
- **Raw EEG Data** (Trujillo, 2020): A dataset where EEG (64 channels, 256 Hz) was recorded during the reported Information-Integration categorization task and the reported multidimensional Rule-Based categorization task. (total time: 34.35 hours)
- **Resting State EEG Data** (Trujillo et al., 2017): A dataset comprising 22 subjects for a resting task of 8 mins with 4 mins of eyes closed and 4 mins of eyes open with 64 EEG channels at 256 Hz using active Ag/AgCl electrodes either mounted in a BioSemi electrode cap or via freestanding electrodes. (total time: 3.04 hours)
- **SEED Series** (Zheng & Lu, 2015; Zheng et al., 2018; Liu et al., 2022): The emotional datasets including SEED (15 subjects), SEED-IV (15 subjects), SEED-GER (8 subjects), and SEED-FRA (8 subjects). All EEG signals (62 channels, 1000 Hz) were recorded with the ESI NeuroScan System in response to videos. (total time: 166.75 hours)
- **Siena Scalp EEG Database** (Detti et al., 2020): A database consisting of EEG recordings (31 channels, 512 Hz) of 14 patients employing EB Neuro and Natus Quantum LTM amplifiers, and reusable silver/gold cup electrodes. (total time: 30.47 hours)

- **SPIS Resting State Dataset** (Torkamani-Azar et al., 2020): A dataset including 10 subjects, 2.5 minutes recording in each state (eyes-closed and eyes-open) prior to a 105-minute session of Sustained Attention to Response Task with fixed-sequence and varying ISIs. Monopolar EEG activity (64 channels, 2048 Hz) was collected via 64 Ag/AgCl active electrodes. (total time: 0.83 hour)
- **Target Versus Non-Target** (Korczowski et al., 2019): A dataset including 50 subjects playing Brain Invaders, a visual P300 Brain-Computer Interface using oddball paradigm with adaptive Riemannian Geometry (no-calibration). EEG signals (32 channels, 512 Hz) were acquired by means of a research-grade amplifier (g.USBamp, g.tec, Schiedlberg, Austria) and the g.GAMMAcap. (total time: 16 hours)
- **TUAR** (Buckwalter et al., 2021): This subset of TUEG contains annotations of 5 different artifacts with EEG recorded (23 channels, 256 Hz). (total time: 92.22 hours)
- **TUEP** (Veloso et al., 2017): This is a subset of TUEG that contains 100 subjects with epilepsy and 100 subjects without epilepsy with EEG recorded (19-23 channels, 256 Hz), as determined by a certified neurologist. (total time: 591.22 hours)
- **TUSZ** (Shah et al., 2018): This corpus has EEG signals that have been manually annotated data for seizure events (start time, stop, channel, and seizure type) with EEG recorded (19-23 channels, 256 Hz). (total time: 1138.53 hours)
- **TUSL** (von Weltin et al., 2017): This is another subset of TUEG that contains annotations of slowing events (23 channels, 256 Hz). This corpus has been used to study common error modalities in automated seizure detection. (total time: 20.59 hours)
- **Self-collected EEG Data** (Jiang et al., 2023; 2021; Luo et al., 2022; Li et al., 2021; Tao & Lu, 2020): We further collect EEG data from more than 140 subjects by ourselves (62 channels, 1000 Hz) with the ESI NeuroScan System. (total time: 342.23 hours)

E VISUALIZATION OF VECTOR-QUANTIZED NEURAL SPECTRUM PREDICTION

We further visualize how the amplitude and phase in the Fourier domain are reconstructed. As depicted in Figure 7, although some details are missing, the overall trend of the amplitude is reconstructed well. In contrast, the reconstruction of the phase is not as good as the amplitude. Nevertheless, it can be seen from Figure 6 that there is still a stable decrease in the reconstruction loss during training, which indicates the discrete codebook does learn high-level information from the Fourier domain.

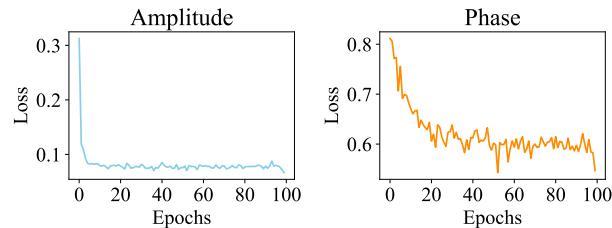


Figure 6: The reconstruction loss curve of amplitude and phase.

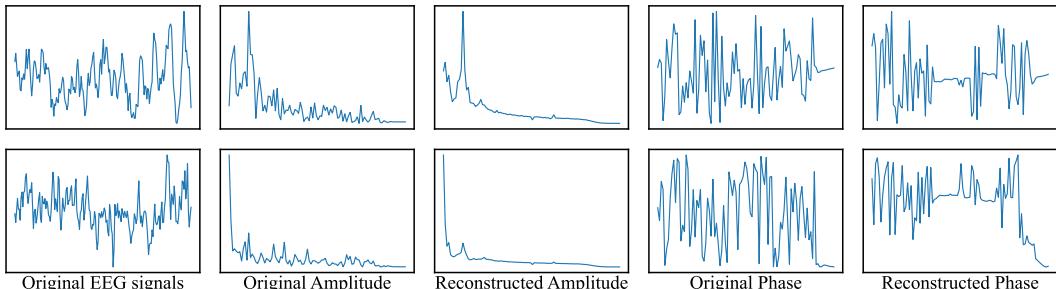


Figure 7: Visualization of reconstructed Fourier spectrum. Note that we only visualize half of the results since DFT is conjugate symmetric.

F MORE EXPERIMENTS ON OTHER BCI TASKS

We conduct two additional BCI tasks on the following datasets:

- **SEED-V** (emotion recognition) (Liu et al., 2021): An emotion EEG dataset containing five emotion categories (happy, sad, neutral, disgust, and fear). The experiment collected EEG data (62 channels, 1000 Hz) from 20 subjects, including 10 males and 10 females. Each subject participated in the experiments three times and each session included fifteen video clips corresponding to the five emotions, where each video clip lasted for several minutes. The EEG signals are segmented into 148,080 1-second samples.
- **MoBI** (gait prediction) (He et al., 2018): A mobile brain-body imaging dataset acquired during treadmill walking in a BCI task, which is a lower limb motor imagery dataset. Six goniometers were employed to record bilateral joint angles on the legs (hip, knee, and ankle). The objective is to regress the angles for 12 targets (left leg and right leg). The data were collected from 8 healthy subjects, each of whom had three identical trials. The EEG signals (60 channels, 100 Hz) were recorded by the ActiCap system. Setting the stride to 50 ms, the dataset involves 575,830 2-second samples.

For SEED-V, as there are fifteen trials for one session, we separate the fifteen trials into three parts with an equal number of trials, i.e., 5:5:5. We merge each part from all sessions of subjects and derive the training, validation, and test set. As SEED-V is overall balanced, we consider accuracy instead of balanced accuracy as a metric to compare performance. Note that some implementation details are a bit different from default settings on this dataset due to different characteristics (peak learning rate: 5e-4 (L) 5e-3 (H); total epochs: 50 (L/H), warmup epochs: 4 (L) 5 (H)).

For MoBI, each trial consisted of a 15-minute treadmill walking session (training session), followed by a 5-minute treadmill walking session (test session) with a closed-loop BCI. To validate the model, we split the training session into two parts: the first 10 minutes of EEG and its corresponding joint data were used as training data, while the last 5 minutes of data were used as validation data. Meanwhile, we combined all the training data, validation data, and testing data of the eight subjects to form corresponding larger training datasets, validation datasets, and testing datasets. Since most angles are typically lower than 90°, the target angles are divided by 90 for normalization. We report the average value of 12 targets for each metric.

As the task of MoBI is regression, we choose the following metrics to evaluate the performance of different methods: 1) **Pearson’s correlation**: Pearson’s correlation coefficient which is used to quantify the models’ regression effect. It measures the linear correlation between two variables X and Y. 2) **R2 score**: R^2 (coefficient of determination) regression score function, which measures how well a statistical model predicts an outcome. 3) **RMSE**: Root Mean Square Error is the standard deviation of the residuals (prediction errors). R2 score is utilized as the monitor to select the best model. MSE loss is the objective to optimize the models.

The experimental results are presented in Figure 6. On SEED-V, LaBraMs outperform all baseline methods on all metrics. The phenomenon that the performance increases when the model gets larger is also observed. For MoBI, our Base model archives competitive results compared to the best baseline method. Whereas, the Large and Huge models obtain better performance among all methods.

Table 6: The results of different methods on SEED-V and MoBI.

	SEED-V			MoBI		
	Accuracy	Cohen’s Kappa	Weighted F1	Pearson’s Correlation	R2 Score	RMSE↓
SPaRCNet	0.2887±0.0047	0.1032±0.0083	0.2904±0.0064	0.4561±0.0161	0.1467±0.0064	0.1344±0.0006
ContraWR	0.3603±0.0098	0.1988±0.0114	0.3590±0.0091	0.3357±0.0164	0.0743±0.0093	0.1401±0.0008
CNN-Transformer	0.3665±0.0058	0.2034±0.0060	0.3638±0.0065	0.3224±0.0109	0.0628±0.0089	0.1411±0.0007
FFCL	0.3686±0.0059	0.2094±0.0078	0.3679±0.0062	0.3158±0.0235	0.0712±0.0124	0.1396±0.0014
ST-Transformer	0.2772±0.0047	0.0783±0.0071	0.2625±0.0061	0.5442± 0.0012	0.2911± 0.0014	0.1222± 0.0001
BIOT	0.3802±0.0094	0.2247±0.0100	0.3809±0.0114	0.2757±0.0173	0.0597±0.0069	0.1401±0.0006
LaBraM-Base	0.4095±0.0062	0.2613±0.0075	0.4120±0.0057	0.5383±0.0102	0.2876±0.0032	0.1225±0.0003
LaBraM-Large	0.4096±0.0075	0.2639±0.0090	0.4127±0.0079	0.5603±0.0020	0.3093±0.0032	0.1197±0.0003
LaBraM-Huge	0.4102±0.0037	0.2646±0.0046	0.4136±0.0047	0.5632±0.0023	0.3145±0.0032	0.1196±0.0003

G EFFECTIVENESS OF VECTOR-QUANTIZED NEURAL SPECTRUM PREDICTION

To verify the effectiveness of vector-quantized neural spectrum prediction, we elaborate on three types of experimental settings as illustrated in Table 7. The comparison between LaBraM and Setting 1 demonstrates that the codebook is effective for masked EEG modeling. LaBraM obtains the best performance on TUEV and the lowest standard deviations on TUAB. There is an interesting observation that masked EEG modeling with the assistance of training an auxiliary neural tokenizer (LaBraM and Setting 1) performs greatly better on TUEV while the naive masked EEG modeling (Setting 2 and Setting 3) performs slightly better on TUAB. One explanation for this phenomenon is that learning semantic representations from the neural tokenizer and codebook significantly benefits high-level downstream tasks like TUEV which classifies different types of events. Whereas, TUAB is a low-level downstream task where the clinically normal/abnormal EEG segments can be easily distinguished visually. Hence, simply reconstructing origin signals or their Fourier spectrum is able to perform well on these low-level tasks but fails to obtain satisfying performance on high-level tasks.

Table 7: Ablations to validate the effectiveness of vector-quantized neural spectrum prediction.

	TUAB			TUEV		
	Balanced Accuracy	AUC-PR	AUROC	Balanced Accuracy	Cohen's Kappa	Weighted F1
LaBraM	0.8140± 0.0019	0.8965± 0.0016	0.9022± 0.0009	0.6409±0.0065	0.6637±0.0093	0.8312±0.0052
Setting 1	0.8058±0.0044	0.8949±0.0037	0.8964±0.0012	0.6162±0.0174	0.6376±0.0168	0.8170±0.0058
Setting 2	0.8261±0.0030	0.9150±0.0016	0.9067±0.0024	0.5630±0.0313	0.5910±0.0156	0.7979±0.0082
Setting 3	0.8166±0.0073	0.9062±0.0029	0.9053±0.0026	0.5730±0.0133	0.5643± 0.0089	0.7819± 0.0040

Setting 1: We directly predict output embeddings of the neural tokenizer by maximizing cosine similarity instead of predicting the discrete neural tokens from the codebook.

Setting 2: We discard the neural tokenizer and directly reconstruct raw EEG patches by minimizing MSE loss.

Setting 3: We discard the neural tokenizer and reconstruct the Fourier spectrum (amplitude and phase) of raw EEG patches by minimizing MSE loss.

H ABLATION ON MASK RATIO

In this experiment, we conduct different settings of the mask ratio to explore its impact. It is noted that we introduce the symmetric masking strategy, so we only need to validate half of the mask ratios. As the mask ratio is set to r , the symmetric masking will mask $1 - r$ proportion of EEG patches. The ablation results are provided in Table 8, where experiments are conducted on TUAB and TUEV. It can be induced that the best mask ratio is 0.4 (0.6) for TUAB and 0.5 (0.5) for TUEV. Moreover, 0.5 (0.5) is the second-best mask ratio for TUAB while the remaining mask ratios are incredibly close. The performance for mask ratios except 0.5 (0.5) is also similar to each other. Notably, the mask ratio of 0.5 (0.5) achieves smaller standard deviations on both TUAB and TUEV. Therefore, we conclude that 0.5 (0.5) is a relatively good mask ratio for the masked EEG modeling of LaBraM pre-training.

Table 8: Performance of different mask ratios.

Mask Ratio	TUAB			TUEV		
	Balanced Accuracy	AUC-PR	AUROC	Balanced Accuracy	Cohen's Kappa	Weighted F1
0.5 (0.5)	0.8140± 0.0019	0.8965±0.0016	0.9022±0.0009	0.6409±0.0065	0.6637±0.0093	0.8312±0.0052
0.4 (0.6)	0.8145±0.0039	0.9083±0.0030	0.9049±0.0038	0.6174±0.0127	0.6123±0.0094	0.8067±0.0059
0.3 (0.7)	0.7994±0.0037	0.8950± 0.0006	0.8974±0.0008	0.6112±0.0216	0.6089±0.0158	0.8068±0.0086
0.2 (0.8)	0.8039±0.0054	0.8990±0.0050	0.9018±0.0023	0.6054±0.0268	0.6050±0.0152	0.8024±0.0089
0.1 (0.9)	0.8022±0.0041	0.8968±0.0010	0.8992± 0.0007	0.6033±0.0264	0.6181±0.0178	0.8134±0.0094

I ABLATION ON SYMMETRIC MASKING

We conduct an ablation study to verify the contribution of the symmetric masking strategy. Table 9 reports the results on TUAB and TUEV. It is obvious that the performance of most metrics decreases by a remarkable margin on both datasets, especially TUEV. Specifically, without symmetric masking, the performance of the base model increases a little bit on TUAB. Nevertheless, the performance decreases in most other scenarios. This is because the data is sufficient for the base model, so the symmetric masking strategy which acts like data augmentation contributes a little to the model training. For larger models like LaBraM-Huge, the symmetric masking improves the downstream performance as it requires more data. This observation indicates that symmetric masking can not only boost the downstream performance but also improve stability and robustness.

Table 9: Ablation study of symmetric masking (SM).

	TUAB			TUEV		
	Balanced Accuracy	AUC-PR	AUROC	Balanced Accuracy	Cohen's Kappa	Weighted F1
LaBraM-Base w/o SM	0.8140±0.0019	0.8965± 0.0016	0.9022± 0.0009	0.6409±0.0065	0.6637±0.0093	0.8312±0.0052
	0.8155±0.0041	0.9077±0.0069	0.9065±0.0034	0.6284±0.0175	0.6279±0.0260	0.8152±0.0105
LaBraM-Large w/o SM	0.8226±0.0015	0.9130± 0.0005	0.9127±0.0005	0.6581±0.0156	0.6622±0.0136	0.8315±0.0040
	0.8198±0.0042	0.9140±0.0007	0.9106±0.0012	0.6548±0.0246	0.6601± 0.0122	0.8319± 0.0034
LaBraM-Huge w/o SM	0.8258±0.0011	0.9204±0.0011	0.9162±0.0016	0.6616±0.0170	0.6745±0.0195	0.8329±0.0086
	0.8247± 0.0010	0.9188± 0.0005	0.9149± 0.0004	0.6261±0.0178	0.6391± 0.0179	0.8152± 0.0085

J LABRAM WITHOUT PRE-TRAINING

In this experiment, we directly train LaBraM on the downstream datasets from scratch without pre-training to validate the effectiveness of the masked EEG modeling pre-training. The steep performance drop demonstrates the usefulness of pre-training, as illustrated in Figure 8.

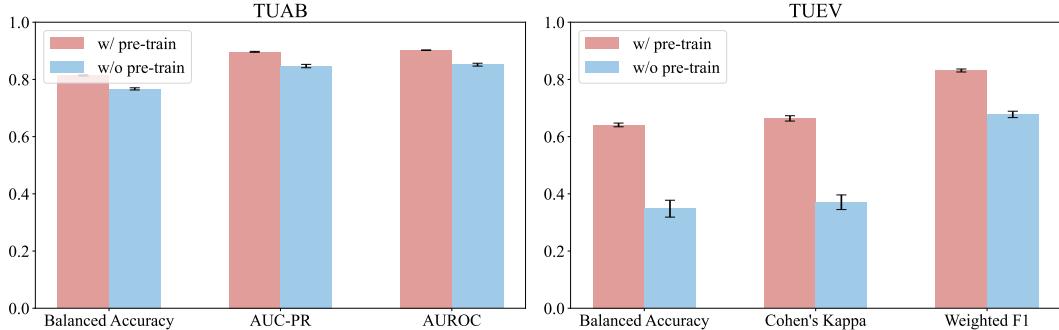


Figure 8: Comparison with model without pre-training.

K PARTIAL FINE-TUNING

In Table 10, we report the results about fine-tuning part of LaBraM. We elaborate on several settings: fine-tuning all 12 Transformer blocks, fine-tuning the last 8 Transformer blocks, fine-tuning the last 4 Transformer blocks, and linear probing. It is noteworthy that for linear probing, we set the weight decay to 0. One can see that on TUAB, the results of full fine-tuning, fine-tuning 12 Transformer blocks, and fine-tuning 8 Transformer blocks are quite similar. When only fine-tuning 4 Transformer blocks and linear probing, there is a slight degradation in performance. On TUEV, yet, fine-tuning 8 Transformer blocks achieves the best performance on all three metrics. Notably, the results of linear probing are much worse than other settings, which still have room for improvement.

Table 10: Results of fine-tuning part of LaBraM.

Fine-tuning Part	TUAB			TUEV		
	Balanced Accuracy	AUC-PR	AUROC	Balanced Accuracy	Cohen’s Kappa	Weighted F1
All	0.8140± 0.0019	0.8965 ±0.0016	0.9022 ± 0.0009	0.6409± 0.0065	0.6637±0.0093	0.8312±0.0052
Transformer (12)	0.8141 ±0.0022	0.8963± 0.0014	0.9022 ± 0.0009	0.6541±0.0250	0.6782±0.0189	0.8386±0.0090
Transformer (8)	0.8134±0.0022	0.8960±0.0019	0.9020± 0.0009	0.6611 ±0.0152	0.6820 ± 0.0089	0.8406 ± 0.0036
Transformer (4)	0.8074±0.0032	0.8930±0.0065	0.8967±0.0018	0.6188±0.0118	0.6560±0.0233	0.8256±0.0114
Linear Probe	0.7954±0.0059	0.8864±0.0030	0.8835±0.0028	0.3461±0.0225	0.3968±0.0329	0.6974±0.0161

L ABLATION ON SPATIAL EMBEDDINGS

The spatial embeddings have helped us address the challenge of heterogeneity in electrode configurations. However, it is important to verify the effectiveness of this approach. During pre-training, we observed that the loss could not converge without spatial embeddings. This was expected, as the model needs spatial embeddings to identify the masked patch to reconstruct. During fine-tuning on downstream datasets, we discard the spatial embeddings and notice a significant drop in performance, as shown in Table 11. This clearly demonstrates the importance of spatial embeddings in capturing spatial information.

Table 11: Ablation study of spatial embeddings (SE).

	TUAB			TUEV		
	Balanced Accuracy	AUC-PR	AUROC	Balanced Accuracy	Cohen’s Kappa	Weighted F1
LaBraM	0.8140 ± 0.0019	0.8965 ± 0.0016	0.9022 ± 0.0009	0.6409 ± 0.0065	0.6637 ± 0.0093	0.8312 ± 0.0052
w/o SE	0.8004±0.0037	0.8922±0.0023	0.8888±0.0018	0.5949±0.0423	0.6069±0.0248	0.8040±0.0111

M DISCUSSION

Limitations. First of all, although we have collected the largest EEG dataset ever of over 2,500 hours and trained the largest model with 369M parameters ever for BCI, it still has a large margin from today’s large vision models and large language models. Our work is only the first step to explore the feasibility of training a large EEG model for learning generic representations. It is delighted to find that training a large EEG model with tremendous EEG data does work and obtain appreciable performance gain compared to existing methods developed for specific BCI tasks. Secondly, LaBraM needs to be fully fine-tuned to adapt to downstream tasks, which might be computation-costly and memory-costly. Finally, LaBraM is trained with unimodal EEG data. It is worthwhile to investigate training large EEG models with other modalities.

Outlook. In view of the above limitations, our paradigm paves the way for further research, encompassing the following aspects: 1) Collecting more EEG data from a variety of BCI tasks, and training a larger EEG model to see whether emergent abilities exist in the EEG model similar to large language models; 2) Leveraging the parameter efficient learning methods, such as adapters, prompt tuning, and LoRA, to reduce the fine-tuning overhead and save space for disks; 3) Incorporating other modalities like image, language, speech, and other physiological signals into large EEG models training to build new paradigms, or aligning EEG representations with other modalities in semantic space, which can be a meaningful and challenging direction for future work.

Title:

GET: A Generative EEG Transformer for continuous context-based neural signals.

Authors:

Omair Ali^{1,4,5}, Muhammad Saif-ur-Rehman^{3,5}, Marita Metzler¹, Tobias Glasmachers², Ioannis Iossifidis³ and Christian Klaes¹

Author affiliation:

¹ Faculty of Medicine, Department of Neurosurgery, University hospital Knappschaftskrankenhaus Bochum GmbH, Germany, ²Institut für Neuroinformatik, Ruhr University Bochum, Germany, ³ Department of Computer Science, Ruhr-West University of Applied Science, Mülheim an der Ruhr, Germany; ⁴Department of Electrical Engineering and Information Technology, Ruhr-University Bochum, Germany; ⁵Institute for Experimental Psychophysiology GmbH, Germany

Corresponding author details:

Omair Ali
omair.ali@ruhr-uni-bochum.de

And

Prof. Dr. Christian Klaes
christian.klaes@gmail.com

Abstract

Generating continuous electroencephalography (EEG) signals through advanced artificial neural networks presents a novel opportunity to enhance brain-computer interface (BCI) technology. This capability has the potential to significantly enhance applications ranging from simulating dynamic brain activity and data augmentation to improving real-time epilepsy detection and BCI inference. By harnessing generative transformer neural networks, specifically designed for EEG signal generation, we can revolutionize the interpretation and interaction with neural data.

Generative AI has demonstrated significant success across various domains, from natural language processing (NLP) and computer vision to content creation in visual arts and music. It distinguishes itself by using large-scale datasets to construct context windows during pre-training, a technique that has proven particularly effective in NLP, where models are fine-tuned for specific downstream tasks after extensive foundational training.

However, the application of generative AI in the field of BCIs, particularly through the development of continuous, context-rich neural signal generators, has been limited. To address this, we introduce the Generative EEG Transformer (GET), a model leveraging transformer architecture tailored for EEG data. The GET model is pre-trained on diverse EEG datasets, including motor imagery and alpha wave datasets, enabling it to produce high-fidelity neural signals that maintain contextual integrity. Our empirical findings indicate that GET not only faithfully reproduces the frequency spectrum of the training data and input prompts but also robustly generates continuous neural signals.

This research highlights the potential of applying cutting-edge generative AI methodologies into BCI frameworks. By adopting the successful training strategies of the NLP domain for BCIs, the GET sets a new standard for the development and application of neural signal generation technologies.

1 Introduction

Generative artificial intelligence (generative AI) is a sub field of AI technology which can produce user specified artificial content including text, images, audio, and synthetic data with the help of generative models [1], [2]. Generative models such as variational autoencoders (VAE) [3] and generative adversarial networks (GANs) [4] have long been at the forefront, allowing for the development of synthetic data samples that exhibit extraordinary diversity and realism [5]. With the advent of diffusion models [6], [7], and the transformer architectures [8], [9], the field of generative AI has revolutionized and made unprecedented progress.

Due to their effectiveness in generating novel and realistic samples, generative AI has found its way in many domains and industries from natural language processing (NLP) [10], [11] to computer vision [12], [13], health care [14], [15], education, research [16], [17] and to art [18]. Initially introduced as a generative model in the domain of NLP, transformers showed that with enough data, it can produce results surpassing other techniques like convolutional neural networks (CNNs). In the NLP domain, pre-trained networks such as Llama 3 and GPT-4 (generative pre-trained transformer) [11], [19] are among the most successful transformer based generative models. The success of generative models in the NLP domain is due to the pre-training of large models (for example GPT-4 and Llama 3) on huge dataset and finetuning them to downstream tasks. Pre-training a model on a big dataset enables it to be utilized as a general-purpose feature extractor, which, after finetuning on downstream tasks with a small dataset, outperforms other models that are typically trained from start for that specific purpose.

Following its success in many other areas, researchers also adopted generative AI in the domain of brain computer interface (BCI) technology. A Brain-Computer Interface (BCI) is a system that allows direct communication between the brain and external devices. By interpreting brain signals, BCIs enable control of computers or machines without physical movement. This technology is crucial for assistive devices, medical rehabilitation, and enhancing human-computer interaction. In the realm of BCI, generative AI is at the frontier of generating synthetic data to augment the training set [20], [21]. Authors in [20] proposed generative model (diffusion model) based data augmentation framework namely Diff-EEG which enables the augmentation of electroencephalography (EEG) signals for detecting Alzheimer's disease. Similarly in [21], authors proposed a framework based on deep convolutional GANs (DCGAN) for generating artificial EEG signals to supplement the training data to improve the classification performance of the classifiers. Generative models such as GANs and VAE have been employed in many studies to enhance the training set to improve the performance of deep learning models for various BCI tasks [22], [23].

Moreover in BCI, generative AI is also gaining significant attention in pattern recognition and predictive modeling tasks [24] and various brain image analysis tasks such as translating neural signals to text (language) or images [25], [26], and segmenting magnetic resonance images (MRI) for identifying traces of lesions after stroke [27]. In [26], functional MRI (fMRI) signals corresponding to neuronal activity in visual and semantic regions of the brain are mapped to image and text components using a generative. Similarly, in [25], a generative model (GPT) is trained to reconstruct semantic language from the fMRI signals. Whereas, in [24],

authors proposed a hybrid architecture based on convolution neural network (CNN) and generative architecture (transformer) namely ConTraNet to classify motor imagery signals in EEG data. In [27], the authors developed a brain image segmentation model namely Consistent Perception GAN (CPGAN), which demonstrated higher segmentation performance over other approaches with less labeled data.

Despite the broad success of generative AI in various BCI domains, research on developing generative models for producing continuous, context-based neural signals remains scant. This process is akin to generating ongoing text in NLP tasks, where continuous language output is derived from an input prompt. The ability to create context-sensitive, continuous neural signals is crucial for advancing BCI research across several areas: simulating brain activity, augmenting data with generated neural signals, utilizing the model for real-time prediction of epilepsy, restoring corrupted data from noisy or faulty recordings, and implementing it as a decoder in neural control tasks. We propose that, within the BCI field, pre-training a large-scale model on diverse datasets using the NLP-inspired context window approach will allow the model to understand and reproduce the context-dependent neural signals. This methodology could pave the way for numerous downstream applications. In this work, we present a pipeline based on a generative model (transformer architecture) named Generative EEG Transformer (GET) for pre-training on EEG data and then generating continuous neural signals based on the input prompt or context window.

2 Materials and Methods

2.1 Data Description

Two different types of neural signals (Motor Imagery (MI) and alpha waves) and two publicly available benchmark datasets are employed to quantify and validate the high generalization capability of the proposed pipeline. These datasets are used for training the generative model GET and employed to provide unseen prompts for neural signal generation. For MI a benchmark dataset containing the data of 9 subjects performing two MI tasks is used [28]. For alpha waves a benchmark dataset from 20 subjects is used [29].

2.1.1 BCI Competition IV dataset 2b (MI-EEG)

The dataset comprises of MI-EEG signals from 9 patients with normal or corrected-to-normal eyesight. Each subject completed five sessions. The MI-EEG data is captured with three bipolar electrodes (C3, Cz, and C4), sampled at 250 Hz, and bandpass filtered from 0.5 Hz and 100 Hz. A notch filter set to 50 Hz is applied. For each trial, the prompts (context input windows) of MI-EEG signal from second 3 to second 5.5 (2.5s in total) containing motor imagining of the hand movement in response to the cue are extracted for training and neural signal generation. The time window corresponding to movement imagination employed here is the same as that used in Ref. [30]. Ref. [28] provides a full description of the dataset.

2.1.2 EEG Alpha waves dataset (alpha-EEG)

This dataset comprises EEG recordings of participants performing a simple resting-state experiment with their eyes open or closed. Data were collected during a pilot experiment held

in the GIPSA-lab in Grenoble, France, in 2017. A total of 20 participants participated in the trial.

EEG signals were recorded with EC20 cap with 16 wet electrodes arranged according to the 10-20 international system. The electrode placements were FP1, FP2, FC5, FC6, FZ, T7, CZ, T8, P7, P3, PZ, P4, P8, O1, Oz, and O2. The reference point was located on the right earlobe and the ground at the AFZ scalp region. The data were collected without using a digital filter and at a sampling rate of 512 samples/second.

Every participant participated in a single session consisting of ten blocks of EEG data recording. Each block was ten seconds long. Each subject was asked to record five blocks with eyes closed (condition 1) and the remaining blocks with eyes open (condition 2). The two circumstances were switched around. The participant was instructed to open or close the eyes in accordance with the experimental condition before the start of each block. The detailed description of the dataset is available in Ref. [29].

2.2 Methods

Here we present the proposed pipeline of GET for simulating and generating the context based neural signals. As shown in **Figure 1**, the proposed GET consists of three blocks namely: the encoder block, the transformer block and the decoder block.

The encoder block encodes the input context window. It then feeds it to the transformer block, where the attention mechanism extracts the short- and long-term dependencies and learns the relations among the samples of each encoded input. The decoder block uses the attention scored input from the transformer block to generate the output sequence.

Encoder block:

The architecture comprises two fully connected linear layers separated by a nonlinear activation function. The first linear layer maps the input signal window into a latent space, serving dual purposes. Primarily, it captures the latent representation of the input signal, essential for extracting vital information necessary for accurate signal representation. Secondly, it facilitates a balance between the breadth of the input signal window and computational efficiency. Expanding the input window enhances the context provided to the model, but also increases computational demands, potentially leading to memory overflow. Conversely, reducing the window size decreases memory usage but at the expense of model performance due to reduced context.

The encoder block addresses this by projecting the input window into a lower-dimensional latent space, thereby managing larger contexts without risking memory overflow by reducing the dimensionality of the input signal window.

The second linear layer operates on the encoded latent space, projecting it linearly into the model dimension d_{model} . For this model, the input window spans 150-time stamps, while the latent dimension is set to 100 to prevent memory overflow and minimize computational costs.

A grid search was performed to determine the optimal batch size of 64 and a model dimension d_{model} of 128.

Transformer block:

The latent space input signal embeddings, generated by the encoder block, serve as inputs to the transformer block. Initially, positional embeddings, matching the model dimension d_{model} , are added to these input embeddings to incorporate positional information. This step is crucial for maintaining the sequential order of events within the neural signal.

The combined embeddings are then processed by the transformer encoder, which comprises seven encoder layers. Each layer features a multi-head self-attention mechanism and a feed-forward block, with layer normalization and residual connections following both components. The multi-head self-attention mechanism is designed to contextually encode the input sequence by calculating the sequence representation across different positions. This method effectively captures the interdependencies among various elements within the same input sequence.

In practical terms, the input embedding vector, combined from the input signal and positional embeddings, is logically divided across multiple ‘heads’ in the self-attention layer. This division allows each segment of the embeddings to concurrently learn different facets or aspects of the input sequence in relation to other sequence elements. Such parallel processing leads to a more nuanced and detailed representation of the input sequence, significantly enhancing the transformer encoder’s ability to capture rich representations.

The transformer encoder layer consists of 6 encoder layers and each multi-head self-attention layer uses 8 heads resembling the original transformer architecture [8]. A feed forward dimension of 256 is used in the feed-forward block of each encoder layer. Dropout is employed to avoid overfitting during training.

Decoder block: Similar to the encoder, the decoder block consists of two fully connected linear layers, separated by a nonlinear activation function. The first linear layer in the decoder block transforms the encoded output from the transformer block, mapping it from the latent space dimension back to the original input space dimension. This adjustment ensures that the output sequence length matches the input sequence length. The second linear layer then processes this transformed input, converting it from the model dimension d_{model} to the output feature dimension.

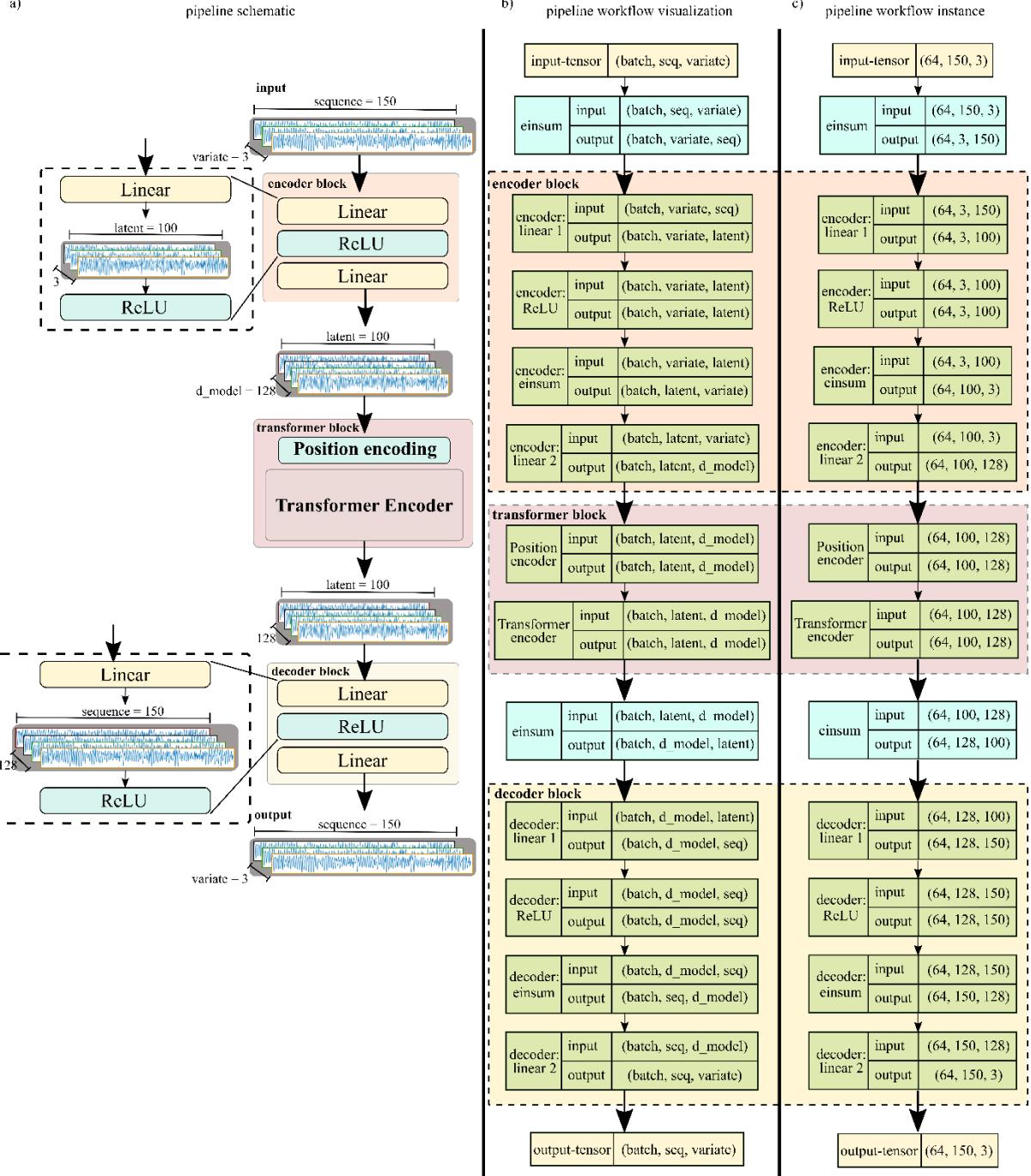


Figure 1: GET. Pipeline for context based neural signal generation. It has three blocks: an encoder block, a transformer block and a decoder block. Input is first fed to encoder block which projects it to latent space and then its embeddings are learned by projecting it to model dimension. The output of the encoder block is the input to the transformer block which learns the relationship between different elements of its input. The output of the transformer block is used as input of the decoder block which maps the input from latent space to the original dimension.

Workflow of information through pipeline: The information workflow through GET as shown in **Figure 1 (b and c)** is as follows:

- **Input:** The input tensor (with dimensions batch, variate and sequence) is fed to the encoder block. Where the variate represents the number of variables (number of electrodes) present in a signal. Univariate refers to signal from single electrode whereas multivariate refers to signals from multiple electrodes. Sequence is the length of input signal which is equal to the input signal window and batch represents the number of samples in single input.
- **Encoder block: (Linear 1):** First fully connected linear layer operates on sequence dimension of the received input tensor and projects it to a latent space with new tensor (of dimensions batch, variate and latent).
- **Encoder block: (ReLU):** Non-linear activation is applied to the output of first encoder linear layer and outputs the same dimensional tensor (batch, variate, latent).
- **Encoder block: (Linear 2):** Second fully connected linear layer learns the embeddings of each sample by projecting it from variate dimension to model dimension d_{model} . The resultant tensor (batch, latent, d_{model}) is fed to the transformer block.
- **Transformer block (Position encoding):** Fixed position encodings of dimension d_{model} resembling the original transformer architecture [8] are added to the received tensor from the encoder block. The resulting tensor (batch, latent, d_{model}) is used as an input to the transformer encoder layer.
- **Transformer block (Transformer encoder):** After coupling the position information with the learned embeddings, the resultant tensor of shape (batch, latent, d_{model}) is fed to the transformer encoder. Therefore, the transformer block outputs the tensor of shape (batch, latent, d_{model}). The details of the functionality of transformer in general and self-attention in particular can be found in [8].
- **Decoder block: (Linear 1):** The output of the transformer block is fed to the decoder block. The first linear fully connected linear layer projects the latent dimension of its input to match the sequence dimension of original input-tensor, thus resulting in the tensor (batch, d_{model} , sequence).
- **Decoder block: (ReLU):** Like the encoder ReLU activation layer, this adds a non-linearity to the output and results in a tensor (batch, d_{model} , sequence).
- **Decoder block: (Linear 2):** The second fully connected linear layer operates on the model dimension d_{model} and learns to map from model dimension to output variate dimension. As a results, it produces a tensor (of dimensions batch, sequence and variate).

3 Results

Here, we present a detailed performance evaluation of our proposed pipeline for generating context-based neural signals that simulate brain activity. Our analysis comprises two main methods. First, we analyzed the Fourier spectrum of both the generated signal and the entire training dataset to determine if the trained model accurately learned and replicated the

frequency spectrum. Second, we employed the Short Time Fourier Transform (STFT) to assess whether the generated signal captured the context of the prompt through its temporal variations in frequency.

Given the challenges of learning from neural signals—namely their low signal-to-noise ratio and non-stationary nature—we initiated our evaluations with univariate signal generation, subsequently advancing to multivariate neural signals.

3.1 Univariate neural signal generation

Here, we present the ability of our proposed pipeline in learning the context and generating the corresponding univariate neural signal. In this analysis, we first employed the filtered signals to train the pipeline. We then fed the unseen prompt to the trained model which generates the neural signal based on the prompt. In this experiment we jointly trained the model on two different EEG datasets (MI-EEG and alpha-EEG dataset). Each dataset contributed equally during the training of the model. In the second step, we used the raw neural signals to train the model.

Univariate filtered signal generation: **Figure 2** shows three instances (a, b, and c) of univariate signal generation based on three randomly selected unseen prompts. In **Figure 2**, each column represents an instance or an example. The first row represents the prompt (in blue) and the generated signal (in red). Second and third rows show the frequency spectrum of the generated signal and the entire training data respectively. In this experiment, we used input windows of size 100 samples to train the model, whereas the generated signals have length equal to 600 samples. Rows two and three clearly indicate that, the generated signals simulate the frequency spectrum of the training data thus indicating the learning of frequency distribution.

Similarly, **Figure 3** represents another three instances of univariate signal generation based on randomly selected unseen prompts. Here, these three instances are taken from alpha waves dataset. It is evident from **Figure 3**, that the produced signals follow the frequency distribution of the alpha-EEG dataset. However, in both cases, the frequency amplitude of the generated signals is slightly higher than the average frequency amplitude of the respective signals. One reason could be the that since we used the latent representations of the input signals to train the model to avoid the memory overflow, it lost some information during this transformation. Secondly, since we employed only the mean squared error as the loss function, it becomes non-trivial to optimize for frequency amplitude without customizing the loss function to include the frequency component.

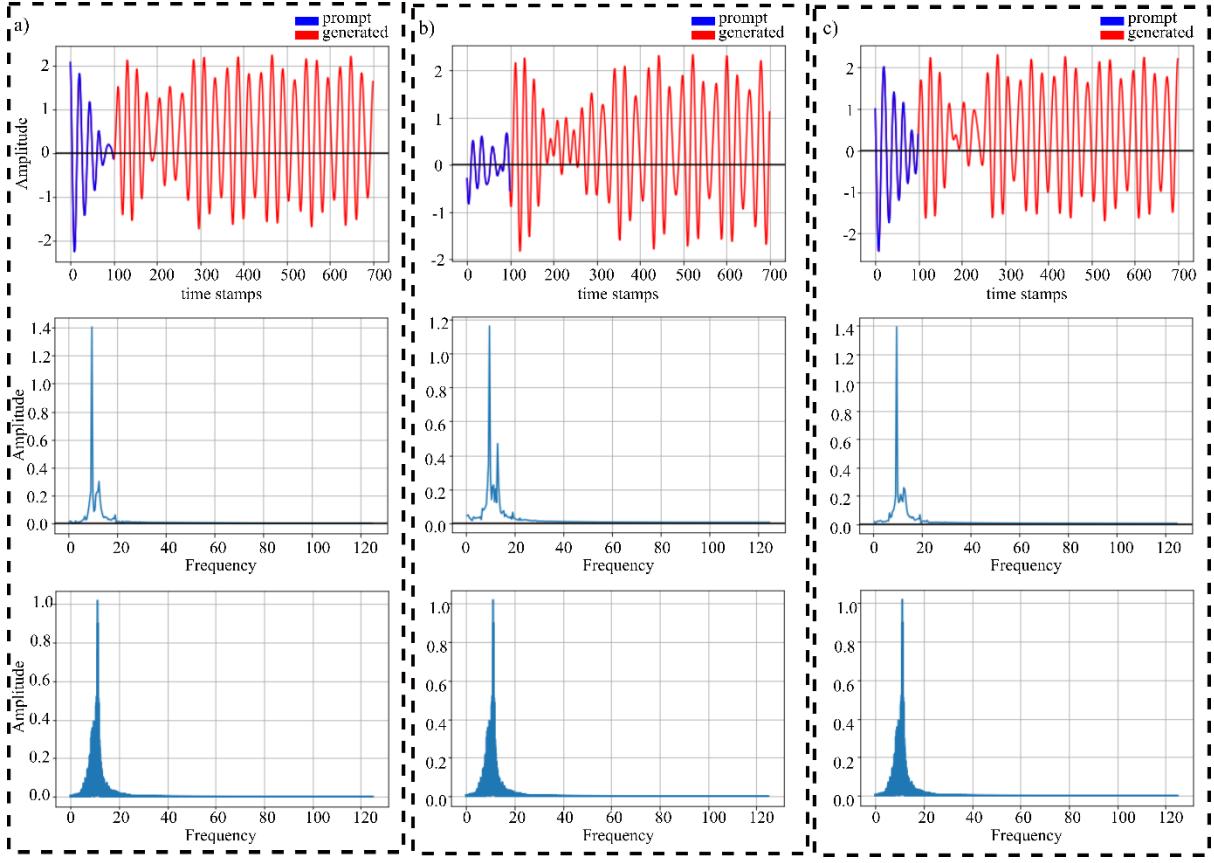


Figure 2: Univariate signal generation of motor imagery EEG signals with prompt of 100 samples. a), b) and c) show three examples of neural signal generation based on randomly selected unseen three input prompts from MI-EEG dataset. For each example, first row shows the input prompt in blue and generated signal in red color. Second row shows the frequency spectra of generated signal whereas the third row shows the frequency spectra of the training data.

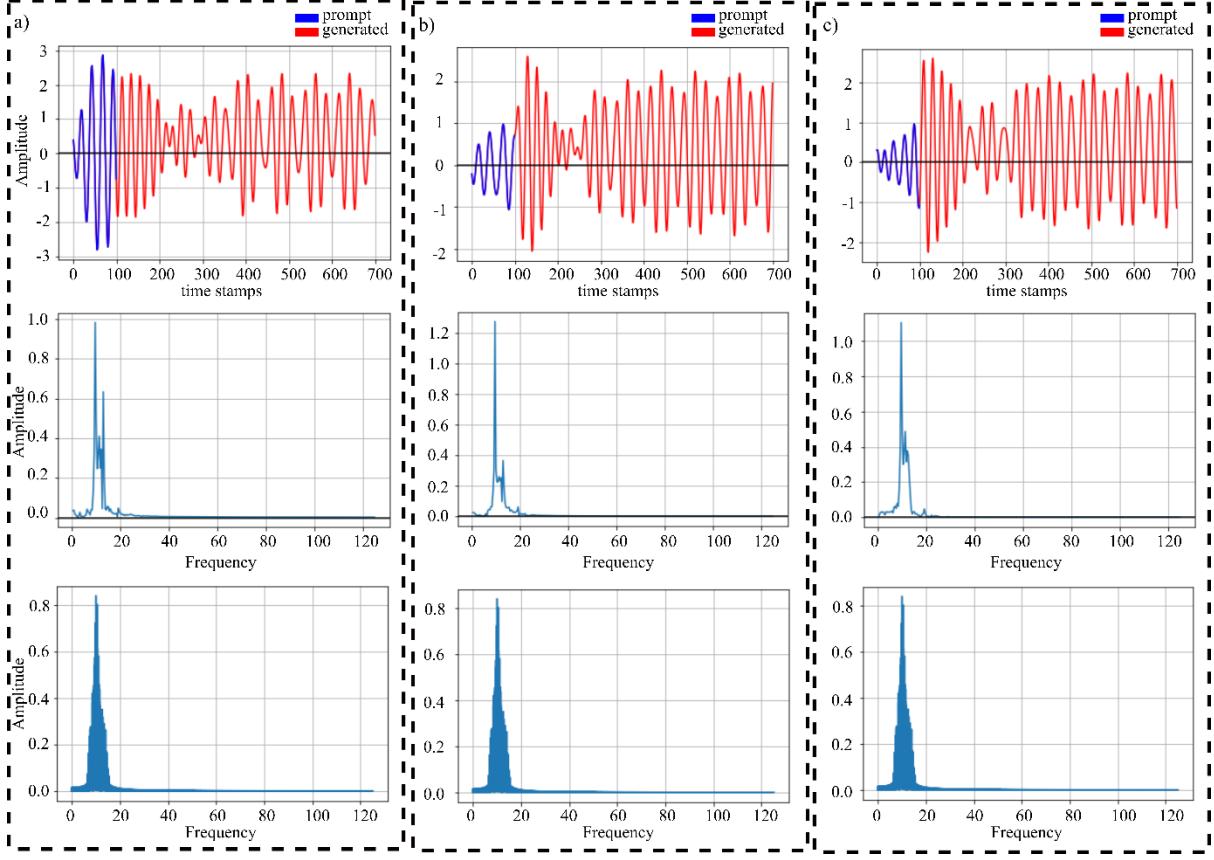


Figure 3: Univariate signal generation of alpha waves neural signals with prompt of 100 samples. a), b) and c) show three examples of neural signal generation based on randomly selected unseen three input prompts from alpha waves dataset. For each example, first row shows the input prompt in blue and generated signal in red color. Second row shows the frequency spectra of generated signal whereas the third row shows the frequency spectra of the training data.

3.1.1 Statistical analysis

In order to find the statistical significance of the generated signals by GET, we computed the power spectral densities (PSD) of the generated signals and compared them with the corresponding ground truth signals. Moreover, we also computed the mean squared error (mse) of the power spectral densities of generated and ground truth signals. PSD is the measure of power content of the signal over its frequency spectrum. It helps to understand how closely the power distribution of the generated signal correlates with that of the ground truth signal. **Figure 4** shows the PSD comparison between the generated and ground truth signals based on three unseen prompts for MI-EEG (a1, a2 and a3) as well as alpha-EEG signals (b1, b2 and b3). It is shown in **Figure 4**, that the signals generated by GET closely follow the power spectral distribution of the ground truth signals indicating the ability of the GET to learn the context from the training data. The mse between the PSD of generated and ground truth signals as reported in **Table 1** quantifies this correlation.

Table 1: Mean squared error (mse) between PSD of generated and ground truth signals.

Mean squared error (mse)	a1) 0.0273	a2) 0.0354	a3) 0.0132
	b1) 0.0034	b2) 0.0029	b3) 0.0038

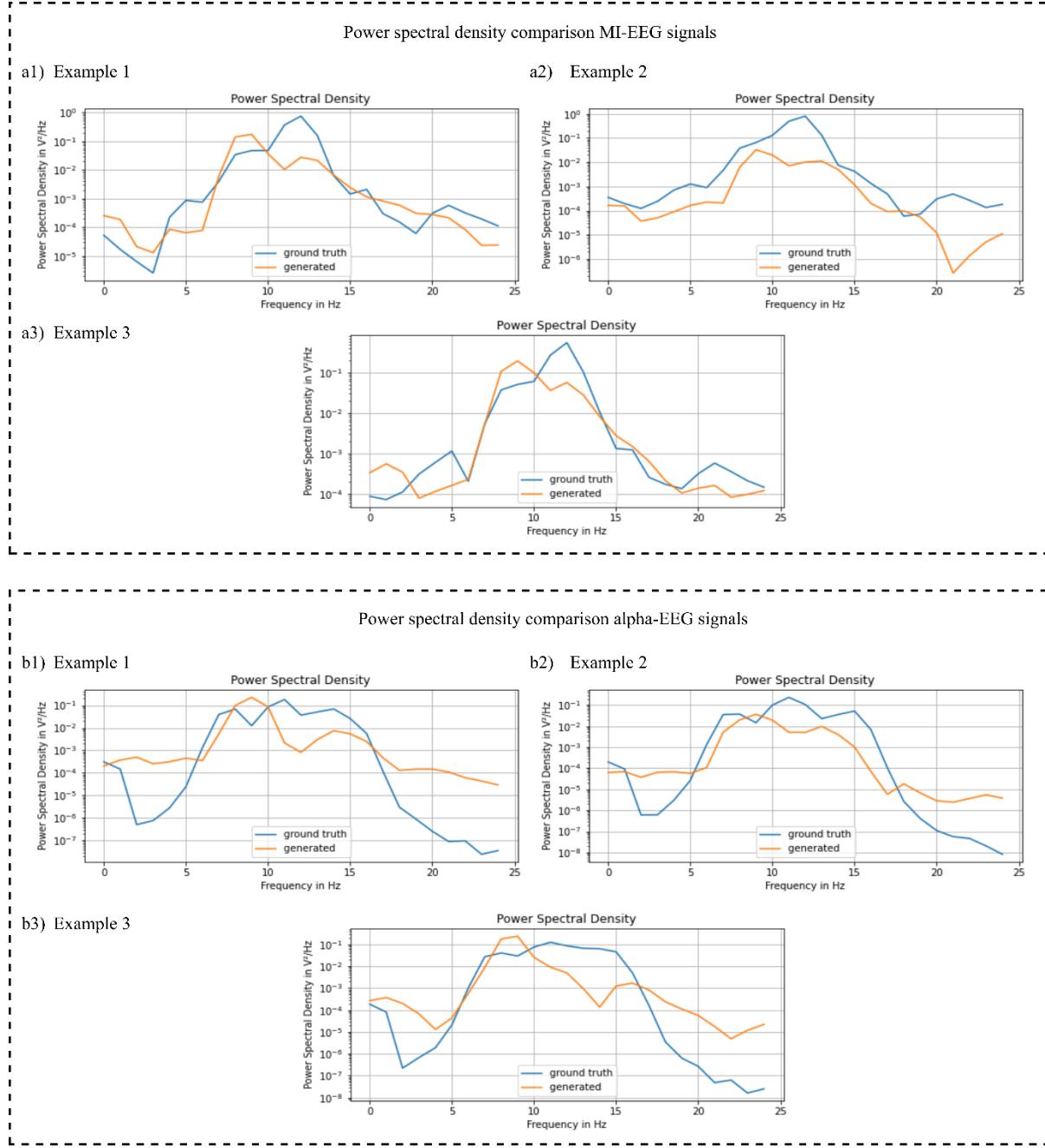


Figure 4: Power spectral density comparison between generated and ground truth signals of MI-EEG signals and alpha-EEG signals. a1,2 and 3) represent the comparison between power spectral densities of generated and ground truth signals based on three randomly selected prompts of MI-EEG signals. b1, 2 and 3 represent the comparison between power spectral densities of generated and ground truth signals based on three randomly selected prompts of alpha-EEG signals.

In another experiment, we jointly trained the model on both datasets using a longer input window with size of 200 samples. In this experiment, both datasets contributed equally during the training process. Here since the context window is doubled compared to previous experiment, we allowed the model to generate longer sequences of size 1000 samples as shown in **Figure 10**.

Figure 10 presents three instances (a, b, and c) of signal generation based on three randomly selected prompts from normal EEG dataset whereas the instances (d, e, and f) show the signal

generation based on prompts taken from alpha waves dataset. It is shown in **Figure 10** (a, b, and c) that model captures the frequency distribution of the normal EEG dataset. However, in this case it also picks up some small magnitude of noise belonging to higher frequency ranges such as between 45-50Hz and 80Hz. Similarly, it is shown in **Figure 10** (d, e, and f) that the model learns the frequency distribution of the alpha waves dataset however, it also produces very small magnitude of noise components in higher frequencies. The noise produced by alpha wave signal generation is smaller in magnitude compared to noise produced by normal EEG signal generation. One probable reason for the generation of noise component is that for longer signal generation, the error accumulates over time which results in these noise components. To mitigate this effect, we retrained the model with higher regularization during training. The result is presented in **Figure 11**. **Figure 11** (a) shows the instance of long sequence generation based on prompt of normal EEG signal, whereas **Figure 11** (b) indicates the signal generation based on alpha wave prompt. In this experiment, we allowed the model to generate even longer sequence of length 1500 samples. Both examples of **Figure 11** show that model learned the frequency distribution of the datasets and the regularization aided to curtail the effect of noise components while producing even longer sequences.

Univariate unfiltered signal generation: In order to evaluate if the generated signals captured the context of the prompt, we performed time frequency analysis to analyze the temporal variations in the frequency spectrum. Henceforth, in this experiment, we computed the spectra of the prompt as well as the generated signal using STFT algorithm. **Figure 5** presents the five instances of neural signal generation based on five randomly selected prompts. Where, each row presents an example of neural signal generation based on the prompt. Second column shows the spectra of the given prompts whereas the third column depicts the spectra of the generated signals. In this and the following experiments employing unfiltered signals, we trained the models using the prompt size of 150 samples which was obtained empirically. Increasing the prompt size in this case adds more context which is prone to more noise which consequently makes it non-trivial to project to latent representation which is then used by the transformer encoder for training. Henceforth, we found empirically the optimal size of input window with 150 samples in case of unfiltered signals.

The prompt in the example displayed in **Figure 5** (a) includes activities in the frequency range of 5-8 Hz, 10-12 Hz, and some activity near 20 Hz. The signal that is generated adheres to the prompt's context and produces comparable activities in those frequencies, along with some extra activity in the sub-5 Hz region. Comparably, in the example shown in **Figure 5** (b), when the prompt exhibits dominant activity in the 10 Hz range, the generated signal also exhibits activity in the 10 Hz region, but it also exhibits activity in the 5-6 Hz range. Given the nearly 2.5-second duration of the generated signal, it is anticipated to elicit activity in other frequency ranges in order to replicate the realistic brain signals.

Similarly, in the example shown in **Figure 5** (d), where the prompt elicited major activities in range 10-12 Hz, the produced neural signal also depicts the activity in this range. However, it also generates strong activities in ranges 5-8 Hz which concords with the activity ranges of the entire training data.

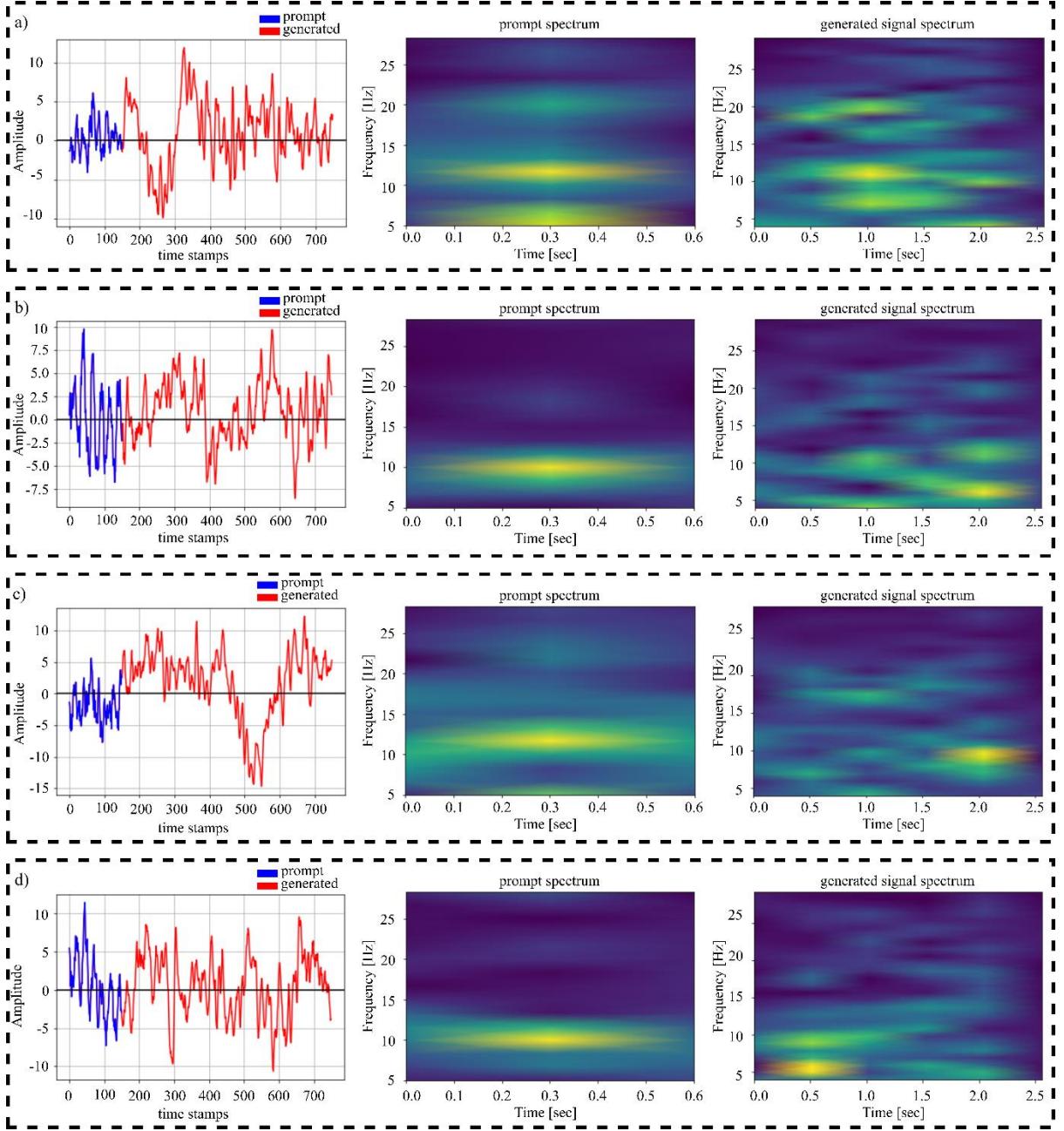


Figure 5: Univariate signal generation of neural signals with unfiltered prompt of 150 samples. a) and b) show two examples of neural signal generation based on randomly selected unseen input prompts of MI-EEG dataset. Whereas c) and d) show two examples of neural signal generation based on randomly selected unseen input prompts of alpha waves dataset.

For each example, first column shows the input prompt in blue whereas the generated signal in red color. The second column shows the frequency spectra of the input prompt generated by STFT whereas the third column represents the frequency spectra of generated signal.

3.2 Multivariate neural signal generation

Here, we demonstrate the scaling ability of our proposed methodology to generate multivariate neural signals simultaneously. For this purpose, we extended the number of electrodes from 1 to 3. The model is trained in similar fashion as in univariate case except for output layer since in multivariate case, it regresses more than one value simultaneously.

Figure 6 shows an example of multivariate neural signal generation. Unlike univariate signal generation, each row here corresponds to neural signal generation of an electrode. Since in this case, we have 3 electrodes, thus three rows. Column 2 shows the spectra of the given prompt of an electrode, whereas column 3 represents the spectra of the generated signals of the respective electrodes.

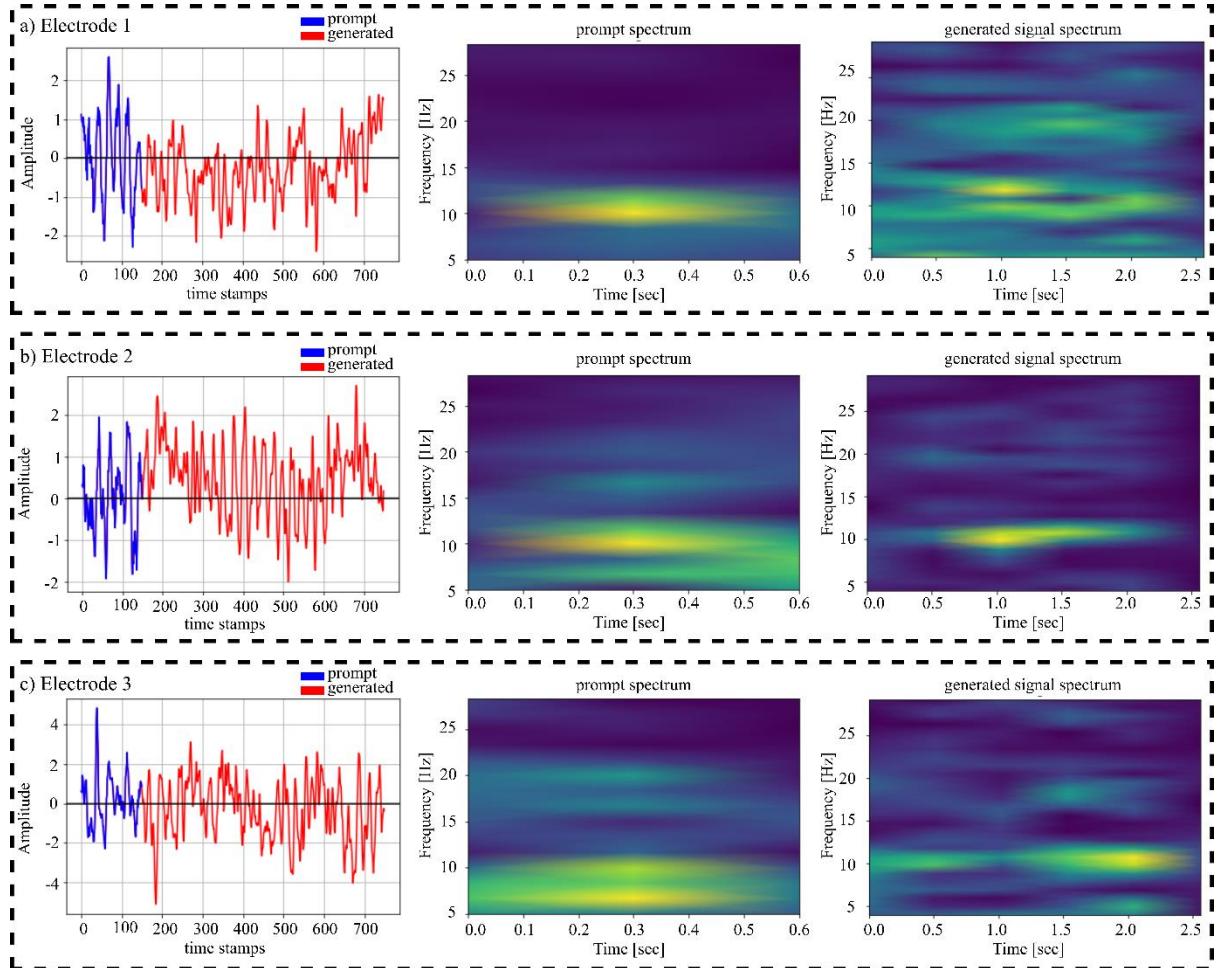


Figure 6: Example 1. Multivariate signal generation of neural signals with unfiltered prompt of 150 samples. a), b) and c) show an example of the multi-electrode neural signal generation based on the unseen prompt from alpha waves dataset. Here each row corresponds to the neural signal generation of the respective electrode. For each electrode, the first column represents the input prompt in blue whereas the generated signal in red color. Columns two and three show the frequency spectra of input prompts and the generated signals respectively.

It is shown in **Figure 6** (a) that prompt of electrode 1 triggered activity in range 10-12 Hz. The corresponding generated signal captures the context of the prompt and produces the similar activity in that range. However, it additionally produces activities in range 12-13 Hz and 20-

22 Hz. However, in case of electrode 2 as shown in **Figure 6 (b)**, the generated signal captured the dominant activity range of the prompt which is around 10 Hz and elicited a strong activity in that range. Contrarily, in case of electrode 3 as presented in **Figure 6 (c)**, where the prompt triggers the activity between 5-10 Hz, the generated signal produced dominant activity near 10 Hz and mild activities near 5 Hz.

In another example shown in **Figure 12**, generated signals capture the context of the respective prompts and produce similar activities. More specifically, prompt of electrode 1 (**Figure 12 (a)**) elicited dominant activity in near 10 Hz which is successfully represented by the model as the generated signal also contains the dominant activity near 10 Hz. Similarly, the prompt of electrode 2 (**Figure 12 (b)**) shows main activity near 10 Hz which is replicated by the neural signal generated in accordance. Furthermore, it also captures the less prominent activity ranges of the prompt including those between 5-8 Hz and produces similar activity in that range. However, the prompt of electrode 3 (**Figure 12 (c)**) elicited activities between 8-10 Hz as well as near 20 Hz. Similar activity is seen in the spectra of the respectively generated neural signal indicating that model captured the context of the prompt.

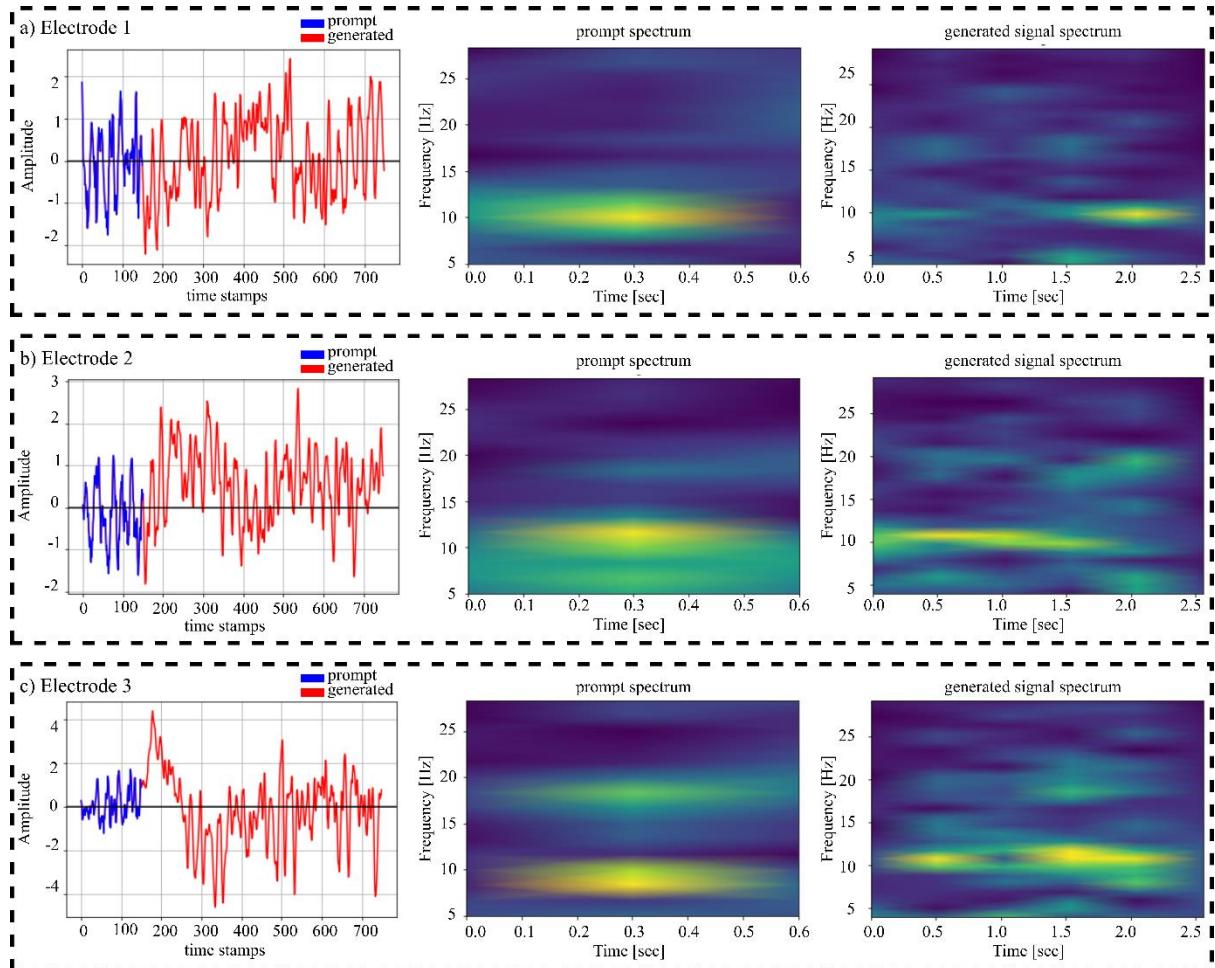


Figure 7: Example 2. Multivariate signal generation of neural signals with unfiltered prompt of 150 samples. a), b) and c) show an example of the multi-electrode neural signal generation based on the unseen prompt from alpha waves dataset. Here each row corresponds to the neural signal generation of the respective electrode. For each electrode, the first column represents the input prompt in blue whereas the generated signal in red color. Columns two and three show the frequency spectra of input prompts and the generated signals respectively.

Figure 7 shows another example of multivariate neural signal generation, where the model learned to extract the prompt’s context and generate neural signals in accordance. Prompts of electrodes 1 and 2 (**Figure 7** (a and b)) generate activities in frequency range 8-12 Hz with more dominant activities close to 10-12 Hz. The generated signals mirror this behavior and produce strong activities near 10 Hz. In case of prompt of electrode 3 (**Figure 7** (c)), the prominent activity lies in range 17-19 Hz and less prominent activity is produced near 10 Hz. This behavior is mirrored by the generated signal as it also elicits dominant activity in range 17-19 Hz and slightly less dominant activity near 10 Hz. This clearly indicates the ability of the learned model to understand the context of the given prompt.

Similarly, another instance of neural signal generation based on context is depicted in **Figure 13**. Here, the prompts of electrode 1 and electrode 2 (**Figure 13** (a and b)) produced prominent activities in range 8-12 Hz which is mirrored by the respective generated signals. The generated signals based on the prompts of electrodes 1 and 2 as shown in **Figure 13** (a and b) also elicited fewer dominant activities in similar range of given prompts. However, for electrode 3, the prompt contained activities in ranges 10-12 Hz as well as near to 20 Hz. The activity pattern is well extracted by the model, and it generated the signal by replicating those activities in the similar ranges. Moreover, it also elicited activities in sub 5 Hz ranges which concord with the frequency spectrum of the training data.

3.2.1 Comparison with ground truth and statistical analysis

In order to evaluate the context learning capability of the GET for multivariate case, it is necessary to find the statistical significance of the generated signals and compare them with the ground truth signals. Henceforth, here we performed the statistical analysis and computed the PSD of the generated signals and compared them with the PSD of the ground truth signals. **Figure 8** shows the comparison between the generated signals and the ground truth signals. Here a1, b1 and c1 show the generated signals and their corresponding spectra based on the given prompts whereas a2, b2 and c2 show the corresponding ground truth signals and their spectra based on the same prompt. It is evident from the **Figure 8**, that signals generated by GET follow the same frequency distribution as that of the ground truth signals. More concretely, in **Figure 9**, PSD comparison between the generated and ground truth signals is presented. Here the PSD is computed by two different methods namely Periodogram and Wlech’s method to get the raw as well as the smooth version of the PSD as shown in a1, b1 and c1. The PSD comparison is shown in a2, b2 and c2. It is evident from the figure that the PSD of the generated signals follow the pattern of the ground truth signals indicating the context learning ability of GET model. To further quantify these results, we also computed the mse between the PSD of generated and the ground truth signals which is shown in **Table 2**.

Table 2: Mean squared error (mse) of the PSD of ground truth and generated signal.

mse	a2) 0.000656	b2) 0.000644	c2) 0.005493
-----	--------------	--------------	--------------

Similarly, another example depicted in **Figure 14** shows the comparison of the generated signals and the corresponding spectra with ground truth signals and their spectra. The statistical analysis as shown in **Figure 15** also represent that the signals generated by GET follow the power distribution of the ground truth signals. This correlation is further quantified by computing the mse between PSD of generated and ground truth signals which is shown in **Table 3**.

The results reported in this, and the aforementioned section shows the ability of our proposed pipeline to learn and extract the context from the prompt and generate signal in accordance.

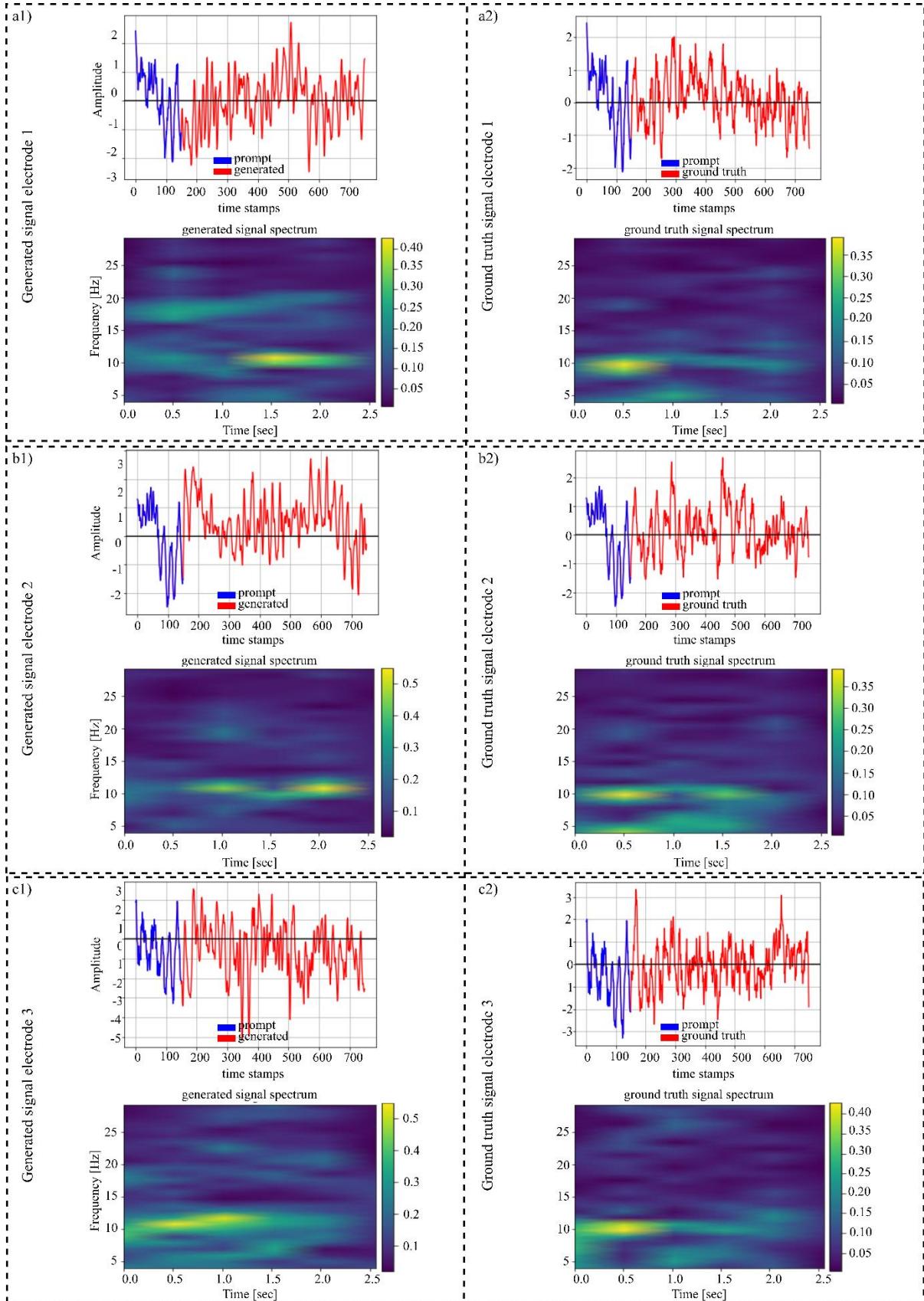


Figure 8: Comparison between generated signals and the ground truth signals based on unseen prompts. a1, b1 and c1 represent the signals generated by GET and their corresponding spectra whereas a2, b2 and c2 represent the ground truth signals and their corresponding spectra.

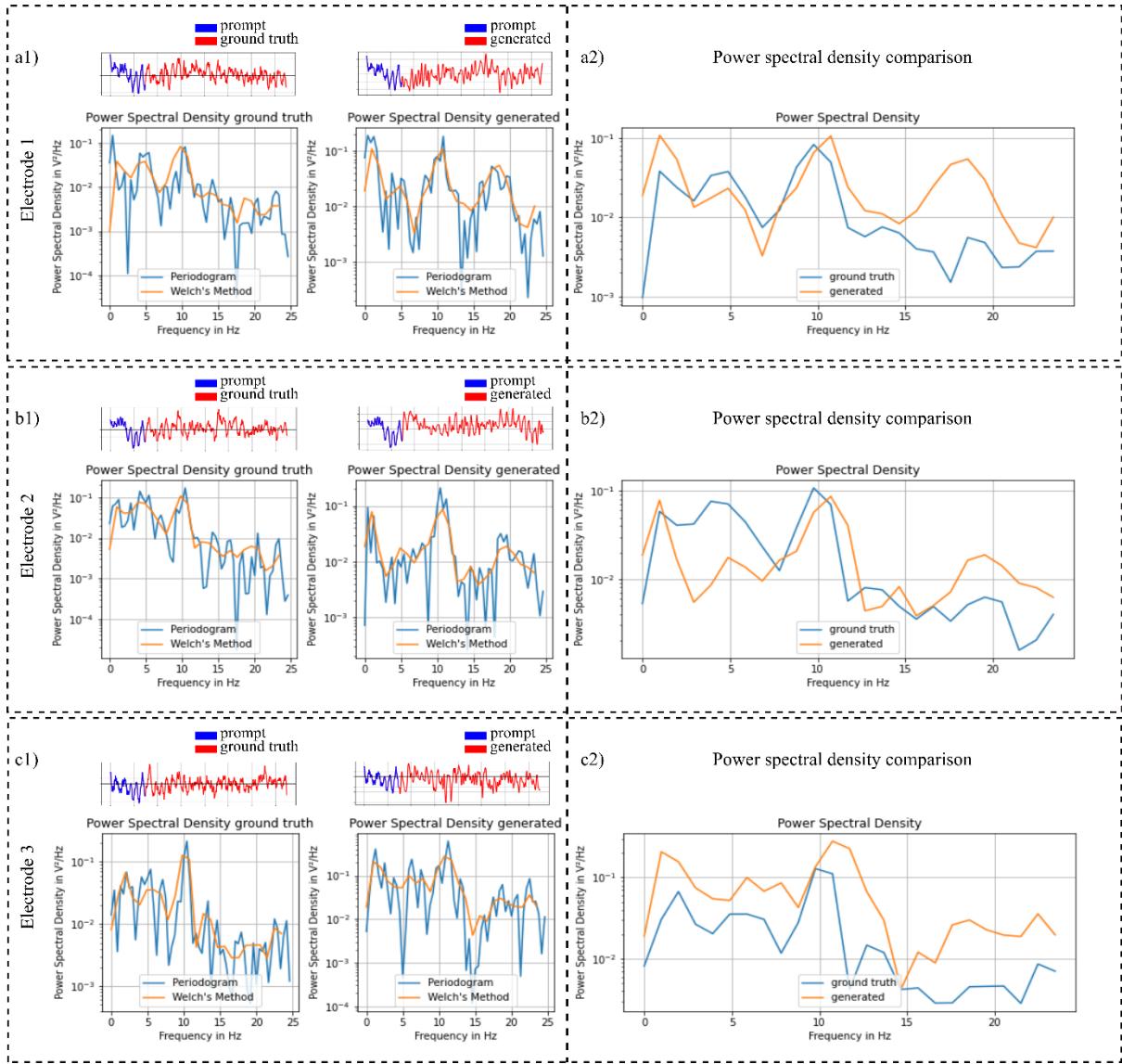


Figure 9: PSD of the generated signals and ground truth signals based on the given prompt. a1, b1 and c1) show the power spectral densities of the generated and ground truth signals computed using two different methods. a2, b2 and c2) show the comparison between the PSD of generated and ground truth signals.

4 Discussion

In this work, we attempted to generate artificial neural signals based on input prompt using proposed GET model. Proposed model successfully learns the context from the prompt (incomplete input) and generate the neural signal similar to the original sample. **Figure 8** shows the input prompt and the complete ground truth EEG signal in **Figure 8** (a2, b2 and c2), and the in **Figure 8** (a1, b1 and c1), shows the corresponding generated EEG signals. Original time series signal and the generated time series signal along with their spectrogram and the power spectral densities show the high generalization quality of GET.

The proposed pipeline of GET consists of three modules namely encoder block, transformer block and a decoder block. One of the intuitions behind employing the stand-alone encoder and decoder block in addition to transformer block is to address the trade-off between performance and memory constraints.

We also evaluated the quality of the generalization of the proposed model based on the length of input prompt. Larger input prompt windows enrich the model (see **Figure 10** and **Figure 11**) with more available context to learn from at the cost of computational expensiveness. On the contrary, feeding smaller input windows may lack the context to extract the meaningful information from them. It consequently can affect the ability of the model to generate the signals in accordance with the prompt. This trade-off is addressed by learning the latent representations of the larger input prompt windows (150 samples) by projecting them to latent space with fewer dimensions (100 samples). The decoder at the end transforms the latent space representation (100 samples) back to its original form (150 samples) thus resulting into generated signal. While the transformer block in between learns the long as well as short term dependencies present in the input signal.

We evaluated the proposed pipeline for generating univariate filtered as well as unfiltered signal based on the given prompt. For this experiment, we trained the model jointly on MI-EEG and alpha-EEG dataset. The results presented in **Figures (2, 3, 4, 5, 10 and 11)** show that the model learned to extract the context from the training data and generalized well on the unseen prompts.

The signals generated based on unseen prompts indicate that the model learned the frequency spectrum from the training data as the frequency spectrum of the generated signals concords with that of the training data. Similarly, the STFT spectra of the unfiltered generated neural signals as shown in **Figure 5** show that the model mirrors the frequency spectrum of the given prompt as well.

Moreover, we also evaluated the GET for generating multivariate neural signals based on the input prompt. In this case, we trained the GET on joint data distribution of MI-EEG and alpha-EEG dataset to generate context based neural signals for three electrodes. The results depicted in **Figures (6, 7, 8, 9, 12 and 13)** show the capability of GET to maintain the context window to learn the context and produce the respective continuous neural signals. As for alpha-EEG signals, where the expected operating frequency is around 8-10Hz during eyes closed task, the model picked the context from its pre-training and generated signals with similar operating frequency as presented in **Figure 6** and **Figure 12**. Similarly for MI-EEG dataset, GET learned

the frequency spectrum from the training data and generated signals in accordance as shown in **Figure 7** and **Figure 13**

We believe that the ability to generate continuous, context-sensitive neural signals is essential for furthering research in BCI in several areas such as brain activity simulation, data augmentation with generated neural signals, epilepsy prediction in real time using the model, data restoration from damaged or noisy recordings, and application of the model as a decoder in neural control tasks. To the best of our knowledge, we have proposed the first generative machine learning model for generating neural signals based on the given neural prompt. This work has the potential to lay the foundation to solve many aforementioned research challenges faced by BCI community.

5 References

- [1] C. Ebert and P. Louridas, "Generative AI for Software Practitioners," *IEEE Softw.*, vol. 40, no. 4, pp. 30–38, Jul. 2023, doi: 10.1109/MS.2023.3265877.
- [2] C. Guan, D. Ding, P. Gupta, Y.-C. Hung, and Z. Jiang, "A Systematic Review of Research on ChatGPT: The User Perspective," in *Advances in Digital Crime, Forensics, and Cyber Terrorism*, N. Mateus-Coelho and M. Cruz-Cunha, Eds., IGI Global, 2023, pp. 124–150. doi: 10.4018/978-1-6684-8422-7.ch007.
- [3] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," 2013, doi: 10.48550/ARXIV.1312.6114.
- [4] I. Goodfellow *et al.*, "Generative Adversarial Nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2014. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf
- [5] G. Raut and A. Singh, "Generative AI in Vision: A Survey on Models, Metrics and Applications," 2024, doi: 10.48550/ARXIV.2402.16369.
- [6] J. Ho, A. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models," 2020, doi: 10.48550/ARXIV.2006.11239.
- [7] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-Based Generative Modeling through Stochastic Differential Equations," 2020, doi: 10.48550/ARXIV.2011.13456.
- [8] A. Vaswani *et al.*, "Attention Is All You Need," 2017, doi: 10.48550/ARXIV.1706.03762.
- [9] M. Chen *et al.*, "Generative pretraining from pixels," presented at the International Conference on Machine Learning, PMLR, Nov. 2020, pp. 1691–1703.
- [10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North*, Minneapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4171–4186. doi: 10.18653/v1/N19-1423.
- [11] T. B. Brown *et al.*, "Language Models are Few-Shot Learners," 2020, doi: 10.48550/ARXIV.2005.14165.
- [12] A. Radford *et al.*, "Learning Transferable Visual Models From Natural Language Supervision," 2021, doi: 10.48550/ARXIV.2103.00020.
- [13] A. Dosovitskiy *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," 2020, doi: 10.48550/ARXIV.2010.11929.
- [14] R. K. Garg, V. L. Urs, A. A. Agrawal, S. K. Chaudhary, V. Paliwal, and S. K. Kar, "Exploring the role of ChatGPT in patient care (diagnosis and treatment) and medical research: A systematic

- review," *Health Promot. Perspect.*, vol. 13, no. 3, pp. 183–191, Sep. 2023, doi: 10.34172/hpp.2023.22.
- [15] C. K. C. Ng, "Generative Adversarial Network (Generative Artificial Intelligence) in Pediatric Radiology: A Systematic Review," *Child. Basel Switz.*, vol. 10, no. 8, p. 1372, Aug. 2023, doi: 10.3390/children10081372.
- [16] Z. Bahroun, C. Anane, V. Ahmed, and A. Zacca, "Transforming Education: A Comprehensive Review of Generative Artificial Intelligence in Educational Settings through Bibliometric and Content Analysis," *Sustainability*, vol. 15, no. 17, p. 12983, Aug. 2023, doi: 10.3390/su151712983.
- [17] M. Imran and N. Almusharraf, "Analyzing the role of ChatGPT as a writing assistant at higher education level: A systematic review of the literature," *Contemp. Educ. Technol.*, vol. 15, no. 4, p. ep464, Oct. 2023, doi: 10.30935/cedtech/13605.
- [18] M. Civit, J. Civit-Masot, F. Cuadrado, and M. J. Escalona, "A systematic review of artificial intelligence-based music generation: Scope, applications, and future trends," *Expert Syst. Appl.*, vol. 209, p. 118190, Dec. 2022, doi: 10.1016/j.eswa.2022.118190.
- [19] OpenAI *et al.*, "GPT-4 Technical Report." arXiv, 2023. doi: 10.48550/ARXIV.2303.08774.
- [20] T. Zhou, X. Chen, Y. Shen, M. Nieuwoudt, C.-M. Pun, and S. Wang, "Generative AI Enables EEG Data Augmentation for Alzheimer's Disease Detection Via Diffusion Model," in *2023 IEEE International Symposium on Product Compliance Engineering - Asia (ISPCE-ASIA)*, Shanghai, China: IEEE, Nov. 2023, pp. 1–6. doi: 10.1109/ISPCE-ASIA60405.2023.10365931.
- [21] F. Fahimi, Z. Zhang, W. B. Goh, K. K. Ang, and C. Guan, "Towards EEG Generation Using GANs for BCI Applications," in *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, Chicago, IL, USA: IEEE, May 2019, pp. 1–4. doi: 10.1109/BHI.2019.8834503.
- [22] A. G. Habashi, A. M. Azab, S. Eldawlatly, and G. M. Aly, "Generative adversarial networks in EEG analysis: an overview," *J. NeuroEngineering Rehabil.*, vol. 20, no. 1, p. 40, Apr. 2023, doi: 10.1186/s12984-023-01169-w.
- [23] F. Fahimi, S. Dosen, K. K. Ang, N. Mrachacz-Kersting, and C. Guan, "Generative Adversarial Networks-Based Data Augmentation for Brain–Computer Interface," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 9, pp. 4039–4051, Sep. 2021, doi: 10.1109/TNNLS.2020.3016666.
- [24] O. Ali, M. Saif-ur-Rehman, T. Glasmachers, I. Iossifidis, and C. Klaes, "ConTraNet: A hybrid network for improving the classification of EEG and EMG signals with limited training data," *Comput. Biol. Med.*, vol. 168, p. 107649, Jan. 2024, doi: 10.1016/j.combiomed.2023.107649.
- [25] J. Tang, A. LeBel, S. Jain, and A. G. Huth, "Semantic reconstruction of continuous language from non-invasive brain recordings," *Nat. Neurosci.*, vol. 26, no. 5, pp. 858–866, May 2023, doi: 10.1038/s41593-023-01304-9.
- [26] Y. Takagi and S. Nishimoto, "High-resolution image reconstruction with latent diffusion models from human brain activity." Nov. 21, 2022. doi: 10.1101/2022.11.18.517004.
- [27] S. Wang, Z. Chen, S. You, B. Wang, Y. Shen, and B. Lei, "Brain stroke lesion segmentation using consistent perception generative adversarial network," *Neural Comput. Appl.*, vol. 34, no. 11, pp. 8657–8669, Jun. 2022, doi: 10.1007/s00521-021-06816-8.
- [28] M. Tangermann *et al.*, "Review of the BCI Competition IV," *Front. Neurosci.*, vol. 6, 2012, doi: 10.3389/fnins.2012.00055.
- [29] G. Cattan, P. L. C. Rodrigues, and M. Congedo, "EEG Alpha Waves dataset." [object Object], Dec. 17, 2018. doi: 10.5281/ZENODO.2348892.
- [30] O. Ali, M. Saif-ur-Rehman, S. Dyck, T. Glasmachers, I. Iossifidis, and C. Klaes, "Enhancing the decoding accuracy of EEG signals by the introduction of anchored-STFT and adversarial data augmentation method," *Sci. Rep.*, vol. 12, no. 1, p. 4245, Dec. 2022, doi: 10.1038/s41598-022-07992-w.

6 Appendix

6.1 Univariate neural signal generation

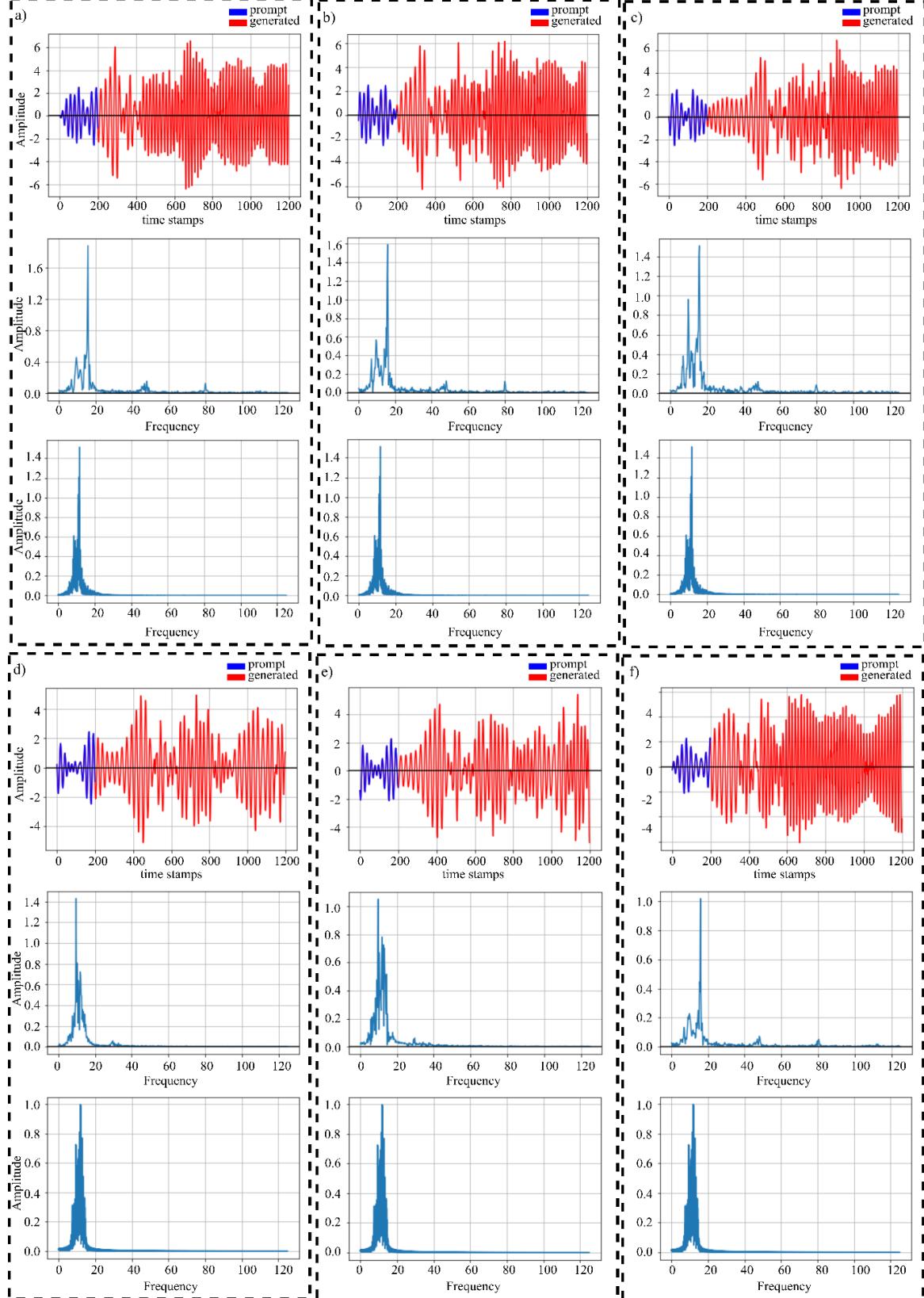


Figure 10: Univariate signal generation of MI-EEG and alpha -EEG signals with prompt window of 200 samples. a), b) and c) show three examples of neural signal generation based on randomly selected unseen three input prompts of MI-EEG

dataset. Whereas d), e) and f) show three examples of neural signal generation based on randomly selected unseen three input prompts of alpha-EEG dataset.

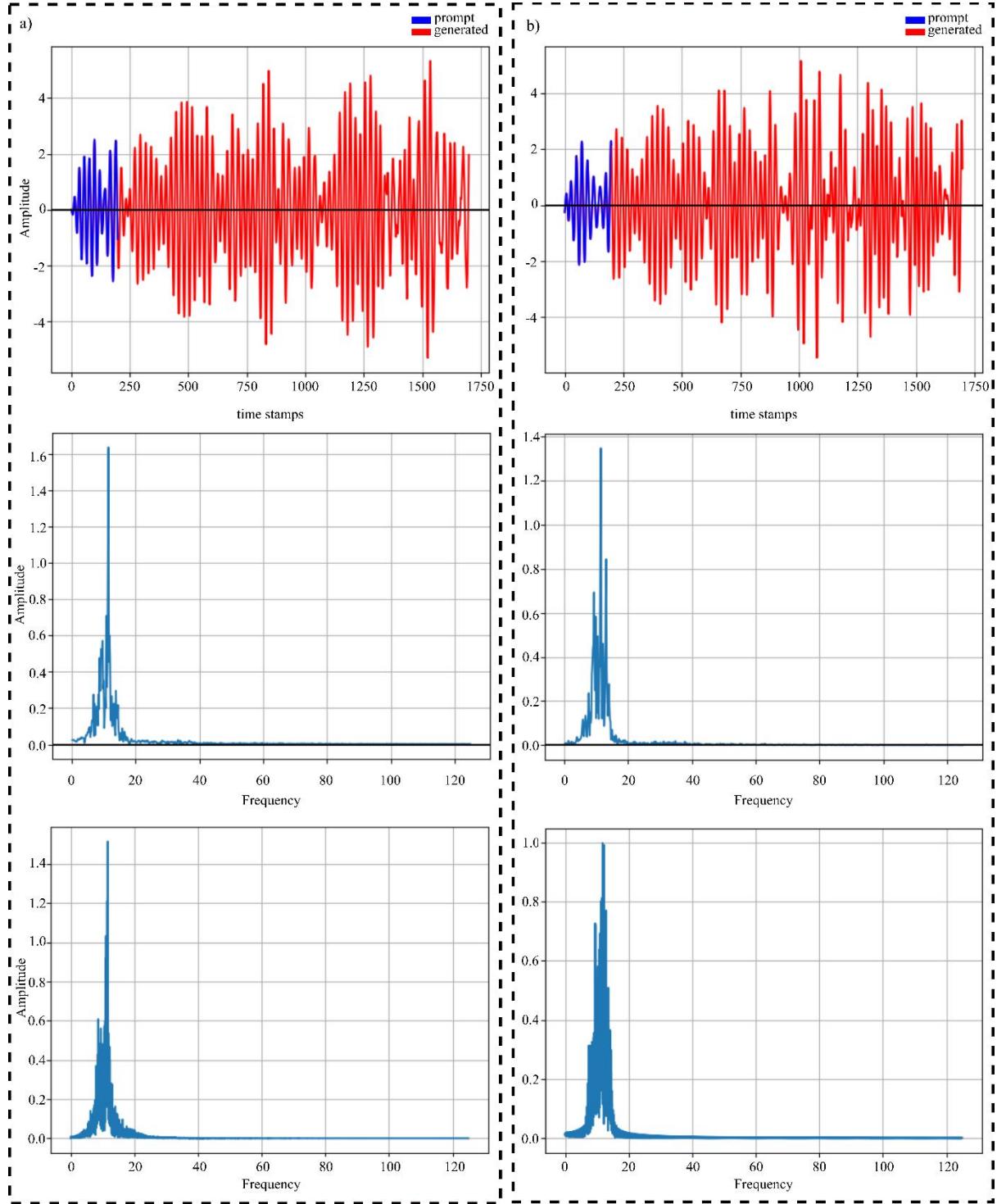


Figure 11: Univariate long sequence signal generation of MI-EEG and alpha-EEG neural signals with prompt window of 200 samples. a) and b) show examples of neural signal generation based on randomly selected unseen input prompts of MI-EEG dataset and alpha-EEG dataset respectively.

6.2 Multivariate neural signal generation

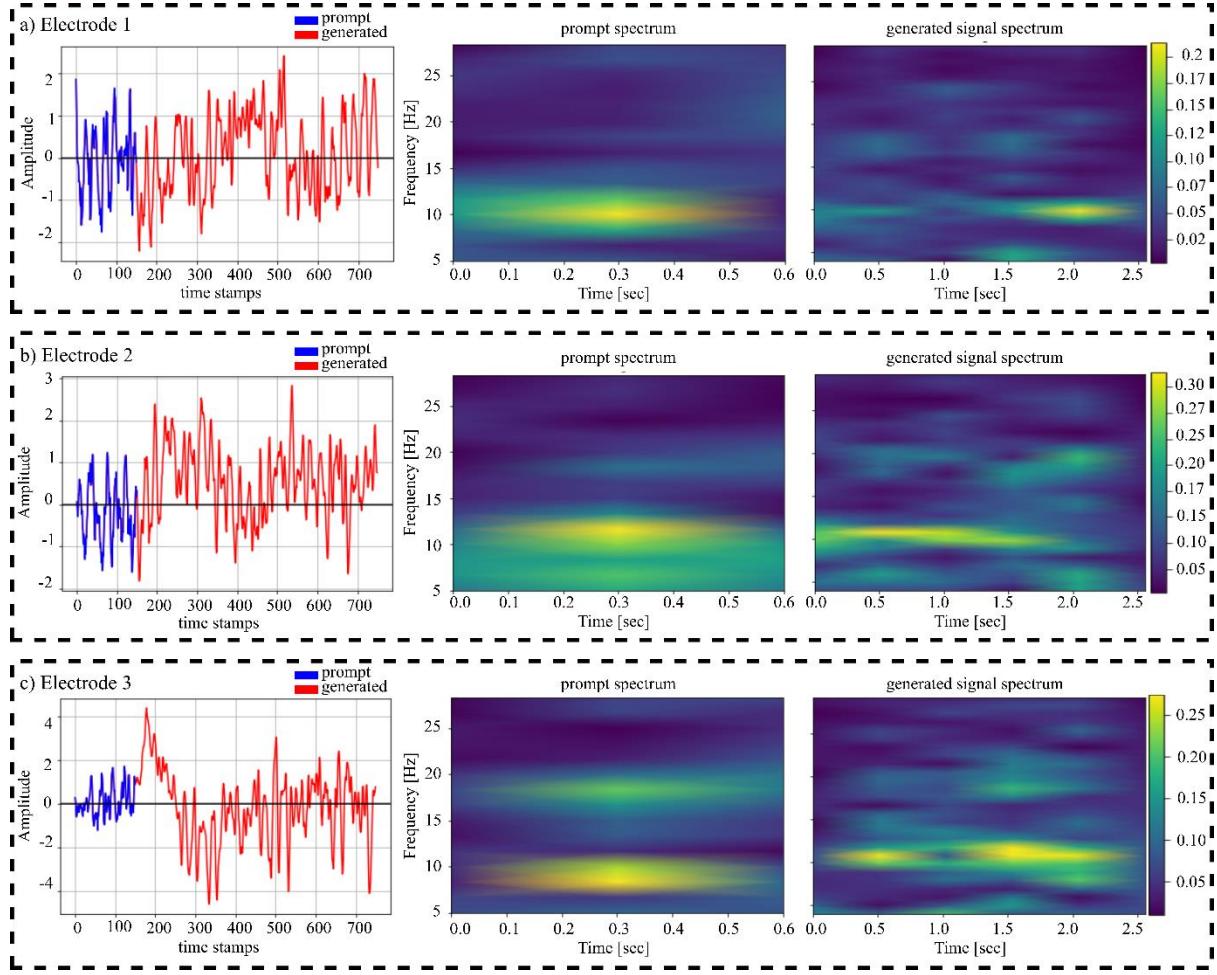


Figure 12: Example 2. Multivariate signal generation of neural signals with unfiltered prompt of 150 samples. a), b) and c) show an example of the multi-electrode neural signal generation based on the unseen prompt from alpha-EEG dataset. Here each row corresponds to the neural signal generation of the respective electrode. For each electrode, the first column represents the input prompt in blue whereas the generated signal in red color. Columns two and three show the frequency spectra of input prompts and the generated signals respectively.

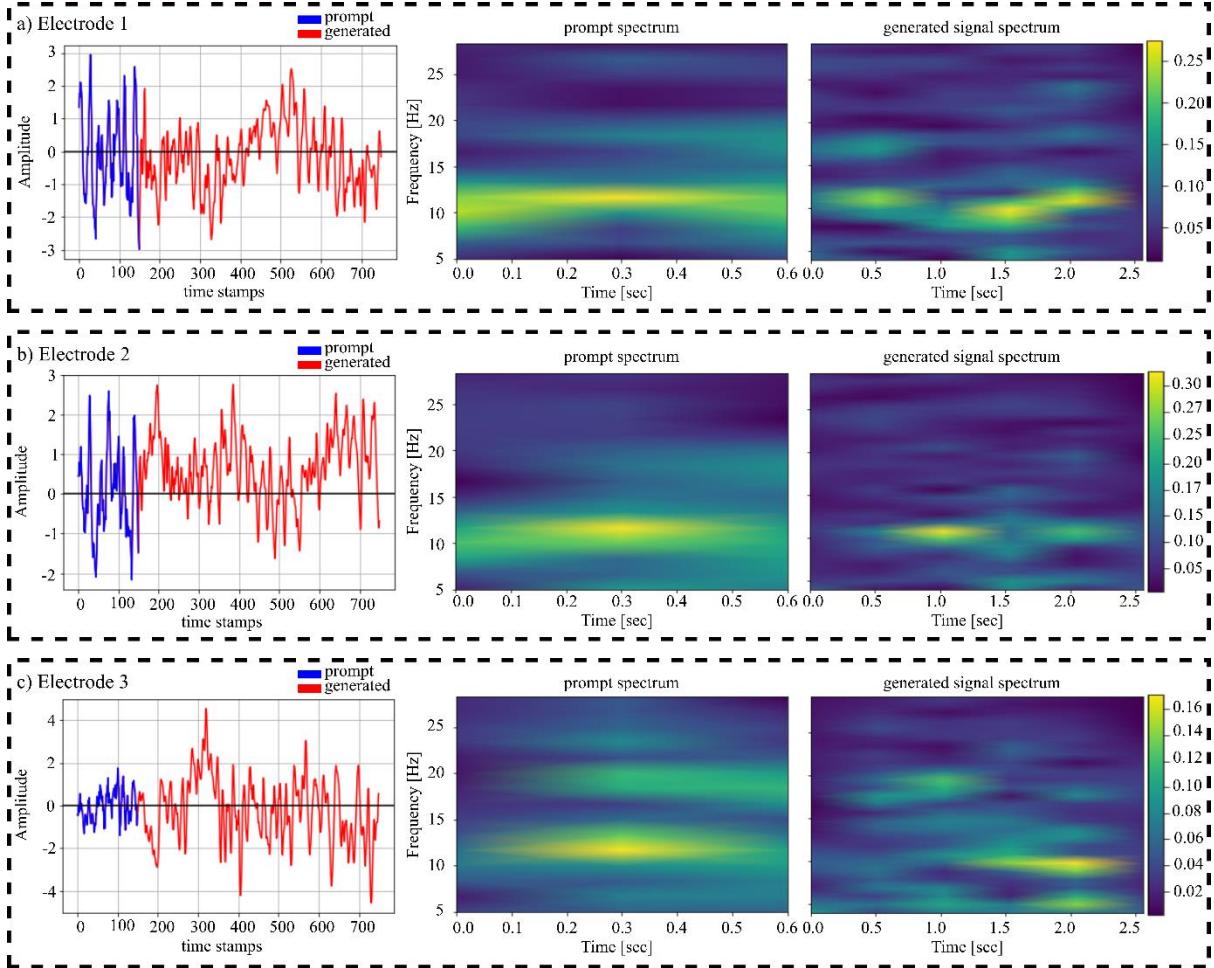


Figure 13: Example 4. Multivariate signal generation of neural signals with unfiltered prompt of 150 samples. a), b) and c) show an example of the multi-electrode neural signal generation based on the unseen prompt from MI-EEG dataset. Here each row corresponds to the neural signal generation of the respective electrode. For each electrode, the first column represents the input prompt in blue whereas the generated signal in red color. Columns two and three show the frequency spectra of input prompts and the generated signals respectively.

6.2.1 Comparison with ground truth and statistical analysis

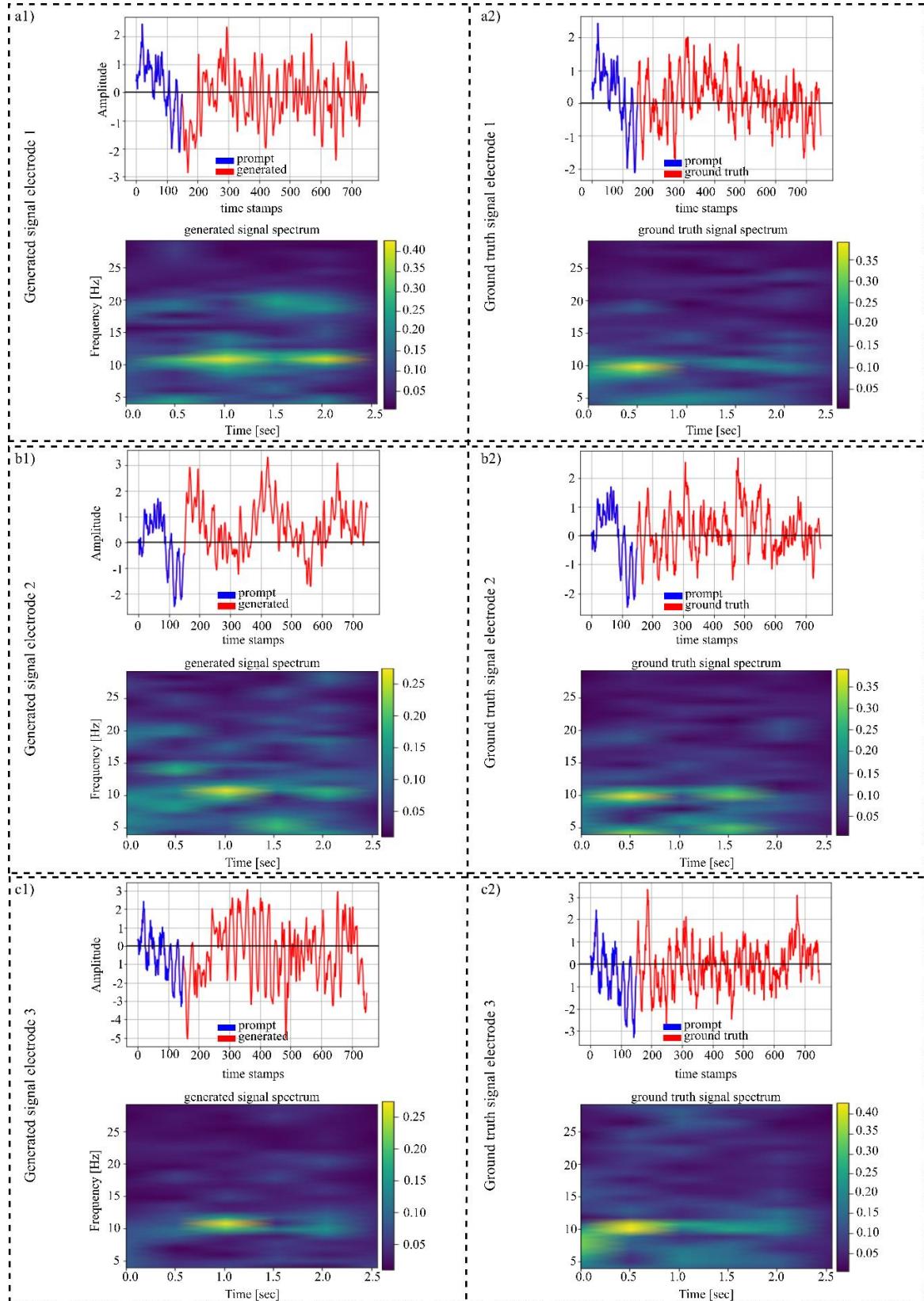


Figure 14: Comparison between generated signals and the ground truth signals based on unseen prompts. a1, b1 and c1 represent the signals generated by GET and their corresponding spectra whereas a2, b2 and c2) represent the ground truth signals and their corresponding spectra.

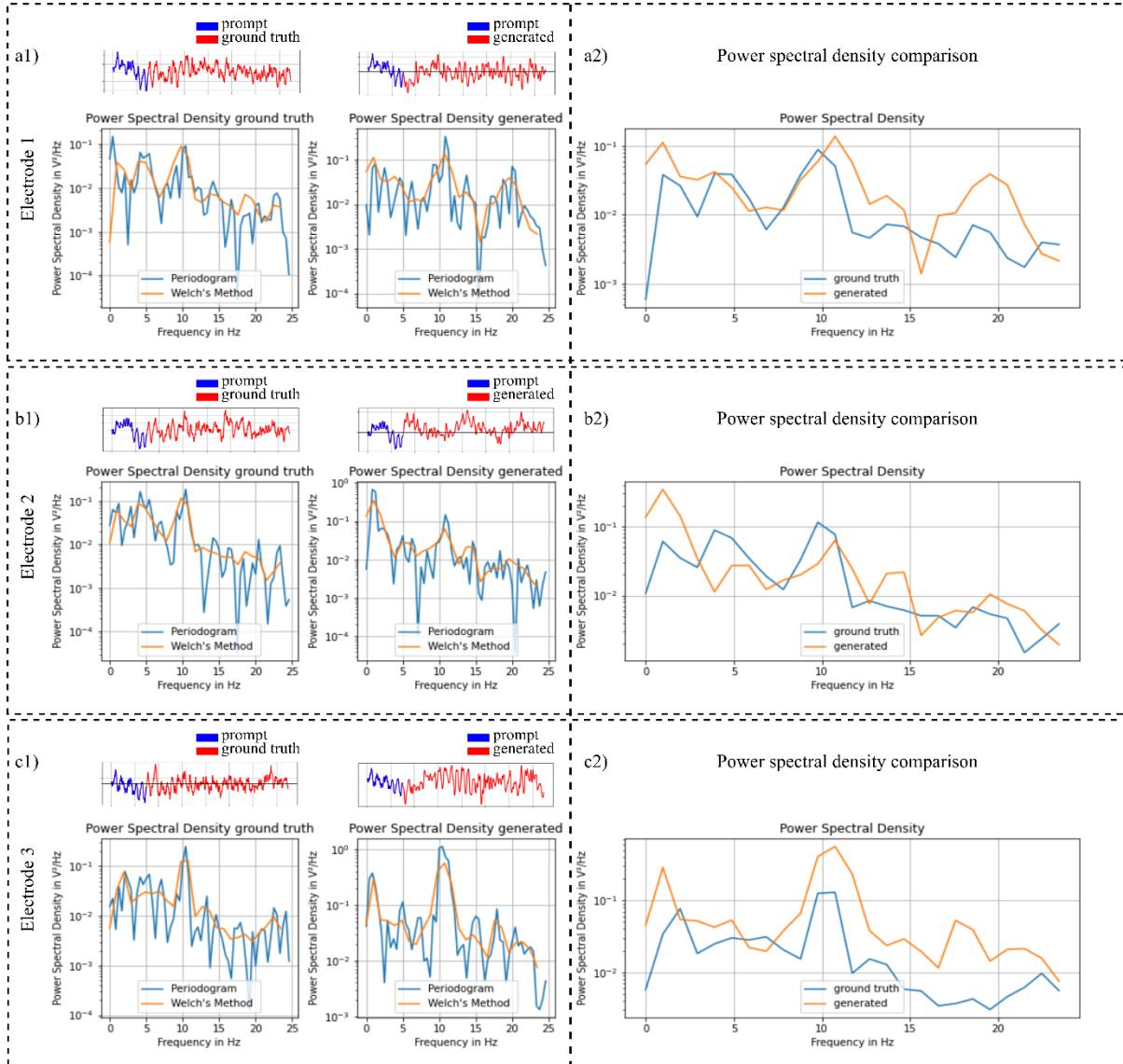


Figure 15: PSD of the generated signals and ground truth signals based on the given prompt. a1, b1 and c1) show the power spectral densities of the generated and ground truth signals computed using two different methods. a2, b2 and c2) show the comparison between the PSD of generated and ground truth signals.

Table 3: Mean squared error (mse) of the PSD of ground truth and generated signal.

mse	a2) 0.000914	b2) 0.004983	c2) 0.015759
-----	--------------	--------------	--------------

EEGMAMBA: BIDIRECTIONAL STATE SPACE MODELS WITH MIXTURE OF EXPERTS FOR EEG CLASSIFICATION

A PREPRINT

Yiyu Gui MingZhi Chen Yuqi Su Guibo Luo[✉] Yuchao Yang[✉]

School of Electronic and Computer Engineering, Peking University

luogb@pku.edu.cn, yuchao.yang@pku.edu.cn

ABSTRACT

In recent years, with the development of deep learning, electroencephalogram (EEG) classification networks have achieved certain progress. Transformer-based models can perform well in capturing long-term dependencies in EEG signals. However, their quadratic computational complexity leads to significant computational overhead. Moreover, most EEG classification models are only suitable for single tasks, showing poor generalization capabilities across different tasks and further unable to handle EEG data from various tasks simultaneously due to variations in signal length and the number of channels. In this paper, we introduce a universal EEG classification network named EEGMamba, which seamlessly integrates the Spatio-Temporal-Adaptive (ST-Adaptive) module, Bidirectional Mamba, and Mixture of Experts (MoE) into a unified framework for multiple tasks. The proposed ST-Adaptive module performs unified feature extraction on EEG signals of different lengths and channel counts through spatio-adaptive convolution and incorporates a class token to achieve temporal-adaptability. Moreover, we design a bidirectional Mamba particularly suitable for EEG signals for further feature extraction, balancing high accuracy and fast inference speed in processing long EEG signals. In order to better process EEG data for different tasks, we introduce Task-aware MoE with a universal expert, achieving the capture of both differences and commonalities between EEG data from different tasks. We test our model on eight publicly available EEG datasets, and experimental results demonstrate its superior performance in four types of tasks: seizure detection, emotion recognition, sleep stage classification, and motor imagery. The code is set to be released soon.

1 Introduction

Electroencephalogram (EEG) is a technique of recording brain activity using electrophysiological indicators, which captures the electrical wave changes during brain activity. EEG can be utilized to detect various human physiological activities such as seizure detection, emotion recognition, motor imagery, sleep stage classification, and other physiological related task Shoeibi et al. [2021], Jafari et al. [2023], Altaheri et al. [2023], Sri et al. [2022].

In recent years, with the development of deep learning, EEG classification models based on deep learning have been widely used Chen et al. [2022]. Among them, models based on Convolutional Neural Networks (CNNs) and Transformers are the most representative, each with their own strengths and weaknesses. CNN-based EEG classification networks have the advantage of faster training and inference speeds, and they perform well on short EEG signals. However, due to the lack of global sequence modeling ability, their performance on long EEG signals cannot be guaranteed Sakhavi et al. [2018], Thuwajit et al. [2021], Schirrmeister et al. [2017]. In contrast, Transformer-based EEG classification networks have good capability of global sequence modeling, achieving excellent performance on both short and long EEG signals. However, as the length of the EEG signal increases, the computational complexity of the model increases quadratically, significantly raising the training and inference costs Dai et al. [2023], Xie et al. [2022], Wang et al. [2022].

Recently, State Space Models (SSM) with selection mechanism and efficient hardware-aware design, such as Mamba Gu and Dao [2023], have shown great potential in long sequence modeling. By utilizing selective state space model, it effectively captures the relationships between tokens in a sequence, addressing the limitation of CNNs in modeling long sequences. Moreover, it exhibits linear computational complexity, which outperforms the quadratic complexity of Transformers and provides a strong backbone network for training EEG classification models on long EEG signals.

Existing EEG classification models always focus on solving specific tasks O’Shea et al. [2020], Phan et al. [2022], Algarni et al. [2022], Autthasan et al. [2021]. However, these networks tend to be less universal across different tasks. While some models consider the generality between EEG tasks, such as EEGNet Lawhern et al. [2018], which has been validated on four tasks including P300 visual-evoked potentials, error-related negativity responses (ERN), movement-related cortical potentials (MRCP), and sensory motor rhythms (SMR), they can only address one type of task in a single training session. Therefore, it is essential to design a classification network capable of handling multi-task EEG data simultaneously.

One of the significant obstacles for multi-task EEG classification is that different EEG data have varying numbers of channels and signal lengths, which makes it difficult for networks to adapt during a single training. For example, MS-HNN Zhu et al. [2023a] is designed for single-channel sleep data and struggles to adapt other multi-channel EEG signals. While MaskSleepNet Zhu et al. [2023b] can classify EEG signals with different numbers of channels by manually setting the channel parameter, it uses a fixed-parameter Multi-scale CNN that can only process EEG signals with limited input lengths. EEG ConvNet Schirrmeister et al. [2017] is designed with a structure capable of adapting to arbitrary signal lengths, it still requires manual setting in different trainings. Therefore, enabling the model to adapt to different signal lengths and channel counts represents a significant challenge.

On the other hand, a network capable of simultaneously handling multi-task EEG data requires a larger network size, more training data, and the ability to address different tasks pertinently. Mixture of Experts (MoE) is a deep learning model with sparse gate-controlled architecture, consisting of a group of expert models and a gating network Jacobs et al. [1991], Shazeer et al. [2016], Xue et al. [2024]. Multiple experts allow for a large increase in the number of model parameters, while the sparse activation mechanism minimizes the impact on the training and inference processes. The gating network can adaptively select experts based on the input, assigning different tasks to different experts, thus achieving task-specificity needed for multitask classification. Therefore, using MoE to achieve EEG multi-task classification might be a feasible solution.

In general, existing EEG classification models mainly face two challenges. On the one hand, these models find it difficult to balance high accuracy and fast inference speed when dealing with long EEG signals. On the other hand, they often struggle to handle different EEG classification tasks and demonstrate poor generality.

To address the aforementioned two issues, we propose EEGMamba, which utilizes bidirectional Mamba suitable for EEG signals, as well as a Spatio-Temporal-Adaptive (ST-Adaptive) module and Task-aware MoE for targeted processing of multi-task EEG classification. Our model enhances Mamba by employing bidirectional modeling to capture the relationships between tokens in a one-dimensional temporal sequence, achieving high accuracy and fast inference speed. Additionally, we propose an ST-Adaptive module that uses spatio-adaptive convolution to process EEG signals of varying channel numbers and a class token to achieve temporal adaptability without any additional processing. To improve generalizability across EEG tasks, we design a task-aware gating network that accurately directs different EEG task tokens to specific experts for processing, while also employing a universal EEG expert to exploit commonalities among different EEG tasks. In summary, our contributions are as follows:

- (1) We design a bidirectional Mamba for EEG signals, which balances fast inference speed with excellent global perception capability.
- (2) We propose a ST-Adaptive module that can automatically adapt to EEG signals of different lengths and channels, thereby processing them simultaneously in single training session.
- (3) We introduce Task-aware MoE with a universal expert, achieving the capture of both differences and commonalities between EEG data from different tasks.

2 Method

EEGMamba primarily consists of the ST-Adaptive module, bidirectional Mamba, and Task-aware MoE. The ST-Adaptive module processes EEG signals of arbitrary lengths and channel numbers through Spatial-Adaptive convolution, Tokenize Layer, and Temporal-Adaptation based on the class token. The features extracted by the ST-Adaptive module are then processed by multiple bidirectional Mamba blocks to perform sequence modeling. Finally, the Task-aware

MoE handles task-specific processing of EEG tokens from different tasks, and a task-aware classifier provides the classification results. The overall model architecture is illustrated in Figure 1.

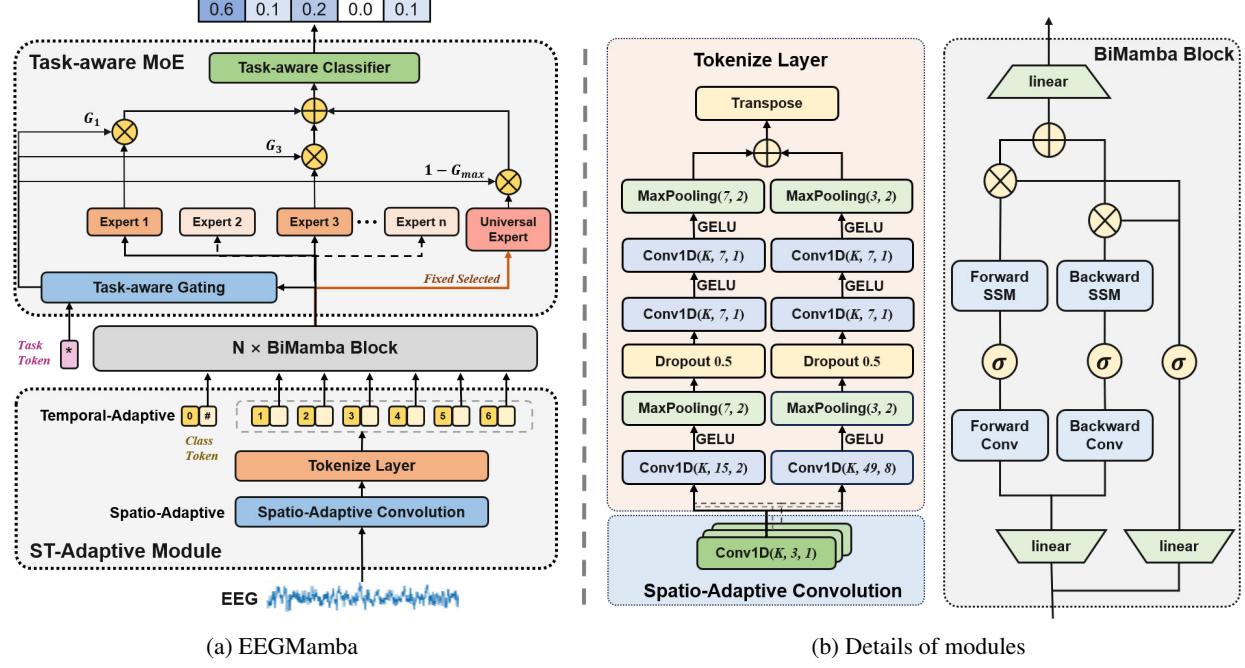


Figure 1: Overall structure of proposed model. The model consists of ST-Adaptive module, Bidirectional Mamba (BiMamba) blocks and Task-aware MoE module.

2.1 Preliminary Work

Mamba is inspired by continuous state-space equations. For continuous input $x(t) \in \mathbb{R}$ in the time domain, the corresponding output $y(t) \in \mathbb{R}$ is determined by the current hidden state $h(t)$ and input $x(t)$ at time t , as shown in Equation (1). Here, $A \in \mathbb{R}^{N \times N}$ is the state matrix, $B \in \mathbb{R}^{N \times 1}$ is related to the system's hidden state, and $C \in \mathbb{R}^{1 \times N}$ is a parameter associated with the input and output.

$$\begin{aligned} h'(t) &= Ax(t) + Bh(t) \\ y(t) &= Ch(t) \end{aligned} \tag{1}$$

Mamba discretizes the continuous-time t into discrete time, transforming the continuous state-space equations into discrete state-space equations. Specifically, by introducing a time-scale parameter Δ , A and B are transformed into discrete-time parameters \bar{A} and \bar{B} respectively. The zero-order hold (ZOH) technique is used as the transformation rule, as shown in Equation (2).

$$\begin{aligned} \bar{A} &= \exp(\Delta A) \\ \bar{B} &= (\Delta A)^{-1}(\exp(\Delta A) - I)\Delta B \end{aligned} \tag{2}$$

In practice, following the approach of Gu et al. Gu and Dao [2023], we approximate \bar{B} using a first-order Taylor expansion, as shown in Equation (3):

$$\bar{B} = (\Delta A)^{-1}(\exp(\Delta A) - I)\Delta B \approx \Delta B \tag{3}$$

Finally, the discretized form of the continuous state space equation is shown in Equation (4).

$$\begin{aligned} h_t &= \bar{A}h_{t-1} + \bar{B}x_t \\ y_t &= Ch_t \end{aligned} \tag{4}$$

Based on the mentioned discrete state-space equations, Mamba further introduces data dependency into the model parameters, enabling the model to selectively propagate or forget information based on the sequential input tokens. In addition, it utilizes a parallel scanning algorithm to accelerate the equation-solving process.

2.2 ST-Adaptive Module

Signals from different EEG datasets often have different lengths and channel numbers. To address this issue, we design a Spatio-Temporal Adaptive module, which converts input signals of arbitrary lengths and channel numbers into uniform features.

We use Spatial-Adaptive convolutional layers to achieve adaptive channel count. Define $x \in \mathbb{R}^{B \times C_i \times L_i}$ as the EEG signal, C_i is the number of EEG channels for the i -th task, and L_i is the length of the EEG signal for the i -th task.

$$y_{SA} = CNN_{SA}(x) \in \mathbb{R}^{B \times D \times L_i} \quad (5)$$

As shown in Equation (5), y_{SA} is the result obtained through Spatial-Adaptive convolution, where the channel dimension is changed from C_i determined by the task to a unified D . Then, y_{SA} is converted into an EEG token sequence through the Tokenize Layer. In order to better extract features from EEG signals, we design a dual-path structure utilizing a small kernel convolution module CNN_S and a wide convolutional module CNN_W . Obtain the small kernel feature token sequence z_s and the wide kernel feature token sequence z_w , respectively. Finally, we concatenate them in the time dimension to form the EEG token sequence T , as shown in Equation (6).

$$\begin{aligned} z_s &= \mathcal{T}(CNN_s(y_{SA})) \in \mathbb{R}^{B \times D \times N_s} \\ z_w &= \mathcal{T}(CNN_w(y_{SA})) \in \mathbb{R}^{B \times D \times N_w} \\ T &= Concat(z_s, z_w) \in \mathbb{R}^{B \times D \times N} \end{aligned} \quad (6)$$

Among them, \mathcal{T} represents the transpose operation, N_s , N_w , N are the number of EEG small kernel feature tokens, EEG wide kernel feature tokens, and overall EEG tokens, respectively. Due to the varying lengths of EEG signals, the number of EEG tokens obtained from the Tokenize Layer is inconsistent.

To achieve temporal adaptation, we introduce a special class token Dosovitskiy et al. [2020]. Specifically, we concatenate this class token with the previously extracted feature token sequence t_s^1, t_s^2, \dots and t_w^1, t_w^2, \dots to obtain the token sequence T^* , as shown in Equation (7).

$$T^* = [t_{cls}, t_s^1, t_s^2, \dots, t_s^{N_s}, t_w^1, t_w^2, \dots, t_w^{N_w}] \in \mathbb{R}^{B \times (N+1) \times D} \quad (7)$$

Then, the input token sequence T^* is processed through a network (using bidirectional Mamba blocks in this study) to integrate EEG token sequence information into the class token. This approach prevents the network from developing biases towards certain tokens in the EEG feature token sequence T due to variations in input length, thereby achieving temporal adaptability.

2.3 Bidirectional Mamba Block for EEG

Mamba is designed for Natural Language Processing (NLP), with its output at each moment depends only on the current input and hidden state, without consideration for future time steps. Since NLP is primarily a generative autoregressive task that relies on previous information for judgment, Mamba's single-directional modeling approach is sufficient to complete such tasks. However, EEG classification tasks require simultaneous processing of both preceding and following information, which cannot be learned by single-directional modeling. Therefore, for EEG signals, the original Mamba's single-directional modeling is insufficient.

To address this issue, we design a bidirectional Mamba for one-dimensional temporal signals, which can model the input bidirectionally and more effectively learn the dependencies between time series tokens. We use the features extracted by the ST-Adaptive module as the input for the bidirectional Mamba.

We denote the input of the bidirectional Mamba block as a sequence T_{k-1} and the output as a sequence T_k . First, T_{k-1} is normalized to T_{k-1}^{norm} by layer normalization. Next, it is mapped by $Linear_X$ and $Linear_Z$ to X_{k-1} and Z_{k-1} , respectively. Then, X_{k-1} enters parallel forward and backward sequence modeling modules. The forward module includes forward 1D causal convolution $Conv_f$ and forward SSM module SSM_f . Similarly, the backward module includes backward 1D causal convolution $Conv_b$ and backward SSM module SSM_b . Then, the results of forward

sequence modeling y_{k-1}^f and backward sequence modeling y_{k-1}^b are summed with Z_{k-1} through gating and then projected through a linear layer Linear to obtain T'_{k-1} . Finally, the output sequence T_k is obtained through residual connection. The detailed process is shown in Algorithm 1:

Algorithm 1 Bidirectional Mamba Block Process

Input: token sequence $T_{k-1} \in \mathbb{R}^{B \times (N+1) \times D}$

Output: token sequence $T_k \in \mathbb{R}^{B \times (N+1) \times D}$

- 1: $T_{k-1}^{\text{norm}} \leftarrow \text{LayerNorm}(T_{k-1})$
 - 2: $X_{k-1} \leftarrow \text{Linear}_X(T_{k-1}^{\text{norm}})$, $Z_{k-1} \leftarrow \text{Linear}_Z(T_{k-1}^{\text{norm}})$
 - 3: $y_{k-1}^f \leftarrow \text{SSM}_f(\text{Conv}_f(\text{Transpose}(X_{k-1})))$
 - 4: $y_{k-1}^b \leftarrow \text{Reverse}(\text{SSM}_b(\text{Conv}_b(\text{Reverse}(\text{Transpose}(X_{k-1}))))))$
 - 5: $T'_{k-1} \leftarrow \text{Linear}(\text{Transpose}(y_{k-1}^f + y_{k-1}^b) \odot \text{SiLU}(Z_{k-1}))$
 - 6: $T_k = T'_{k-1} + T_{k-1}$
-

2.4 Task-aware MoE with Universal Expert

2.4.1 Sparsely-activated MoE

Transformer-based MoE commonly use a sparse activation MoE layer to replace the Feed-Forward Neural Network (FFN) inside the Transformer Fedus et al. [2022]. Each MoE layer consists of several experts, and each expert is typically represented as a Multi-Layer Perceptron (MLP) whose activation is controlled by a gating network Shazeer et al. [2016].

We define N_e as the number of experts, E_i as the i -th expert, and G as the gating network. For each input EEG class token t_{cls}^* , the output y_{cls} of MoE can be expressed as Equation (8):

$$y_{cls} = \sum_{i=1}^{N_e} e_i(t_{cls}^*) * E_i(t_{cls}^*) \quad (8)$$

$$e_i(t_{cls}^*) = \text{Top}_k(G(t_{cls}^*))_i, t_{cls}^* = T_k[0]$$

$$\text{Top}_k(V, k)_i = \begin{cases} v_i, & \text{if } v_i \text{ is top } k \text{ value of } V \\ -\infty & \text{otherwise} \end{cases}$$

2.4.2 Task-aware Gating Networks

A gating network calculates gating values based on the input tokens and selects K experts for activation, typically implemented using a fully connected layer $\text{Linear}_{\text{Gate}}$. However, this can lead to the problem that only a few experts are trained. To avoid this, we adopted the method from Shazeer et al. [2016], adding noise to the gating value computation process using a fully connected layer $\text{Linear}_{\text{Noise}}$, which increases randomness and helps in balancing the load among the experts.

Furthermore, we propose a task-aware gating network which helps improve the accuracy of experts in processing different types of EEG tokens. Specifically, we encode the EEG task into task tokens t_{task} , then concatenate t_{task} with the EEG class tokens t_{cls}^* to obtain t_{cat} , which is then sent to the gating network. The gating values calculated in this manner incorporate task information, allowing for better assignment of different tasks to different experts. The working process of the task-aware gating network is shown in Equation (9), where ϵ represents standard Gaussian noise.

$$t_{cat} = \text{Concat}(t_{cls}^*, t_{task}) \quad (9)$$

$$G(t_{cls}^*, t_{task}) = \text{Linear}_{\text{Gate}}(t_{cat}) + \epsilon * \text{SoftPlus}(\text{Linear}_{\text{Noise}}(t_{cat}))$$

2.4.3 EEG universal expert

EEG signals from different tasks exhibit both differences and commonalities. Only using different experts to process EEG tokens might overlook the connections between tokens from different tasks. Therefore, we design an EEG universal expert that can process EEG tokens from all different tasks and capture their commonalities. To achieve this function, the universal expert is activated for any inputs and not controlled by the gating network's output values.

Overall, our MoE module includes both task experts and universal experts. Task experts can accurately process EEG tokens from different tasks according to gating values, while universal experts can process all EEG tokens. The output of MoE is the weighted sum of these two types of experts. We adopted a weight design scheme similar to Gou et al. [2023], as shown in Equation (10). Here, the output weight ω of the universal expert is determined by the maximum gating value:

$$y = \sum_{i=1}^{N_e} e_i(t_{cls}^*) * E_i^t(t_{cls}^*) + \omega * E^u(t_{cls}^*) \quad (10)$$

$$\omega = 1 - \text{Max}(e(t_{cls}^*))$$

3 Experimental Setup

3.1 Dataset

We evaluate the proposed EEGMamba by using eight datasets from four different tasks, including Bonn Andrzejak et al. [2001], CHB-MIT Shoeb [2009], SleepEDF-20 Kemp et al. [2000], SHHS Goldberger et al. [2000], DEAP Koelstra et al. [2011], SEED Duan et al. [2013], Shu Ma et al. [2022], and BCI-IV-2a Brunner et al. [2008]. Table 1 provides an overview of each dataset. The number of subjects, the number of classes, and the number of channels often varies for different tasks. More details about the datasets can be found in the appendix A.2.

Table 1: Dataset Introduction. ‘# Subjects’ represents the number of subjects, and the same is true for ‘# Classes’ and ‘# Channels’. For the SHHS dataset, we select data from 392 subjects out of 6441 subjects Fonseca et al. [2016].

Dataset	Task	# Subjects	# Classes	# Channels	Sampling Frequency
Bonn	Epilepsy detection	10	5	1	173.61
CHB-MIT	Epilepsy detection	22	2	23	256
SleepEDF-20	Sleep stages classification	20	5	1	100
SHHS	Sleep stages classification	329 from 6441	5	1	125
DEAP	Emotion recognition	32	2	4	128
SEED	Emotion recognition	15	3	62	200
Shu	Motor imagery	25	2	32	250
BCI-IV-2a	Motor imagery	9	4	22	250

3.2 Implementation Details

In the EEGMamba experiment, we train for 100 epochs. The number of bidirectional Mamba blocks and hidden channels is set to 8 and 256, respectively. We use 8 task experts and one universal expert, 2 experts are activated at a time among regular experts. In addition, to demonstrate the effectiveness of the Mamba-based model, we also conduct EEGMamba experiments for each single dataset. In the following text, Single-task EEGMamba is used to represent this experiment. In this experiment, We train for 200 epochs. The number of bidirectional Mamba blocks and hidden channels is set to 2 and 128, respectively. For all experiments, we set the batch size to 128 and the learning rate to 2e-4. The training and test sets are divided in an 8:2 ratio. All models are trained on Intel(R) Xeon(R) Gold 6342 CPU and Nvidia A100 GPUs 80G.

4 Results and Discussion

4.1 Single-task EEGMamba Performance Comparison

We compare the performance of EEGMamba with previous classification models EEGNet Lawhern et al. [2018], Attnsleep Eldele et al. [2021], and EEG Conformer Song et al. [2022] on eight datasets and evaluated them using accuracy (ACC), Area Under Curve (AUC), and F1-score. Figure 2 illustrates the performance comparison of various classification models on different datasets. Obviously, EEGMamba outperforms the other three classification networks across all evaluation metrics in most datasets. It is worth noting that on the CHB-MIT dataset, the extremely imbalanced distribution of seizure and non-seizure samples might make accuracy a less appropriate metric, while F1-score performs

better in reflecting the model’s performance. Our model, particularly, shows a significant advantage over other models in the F1 evaluation metric.

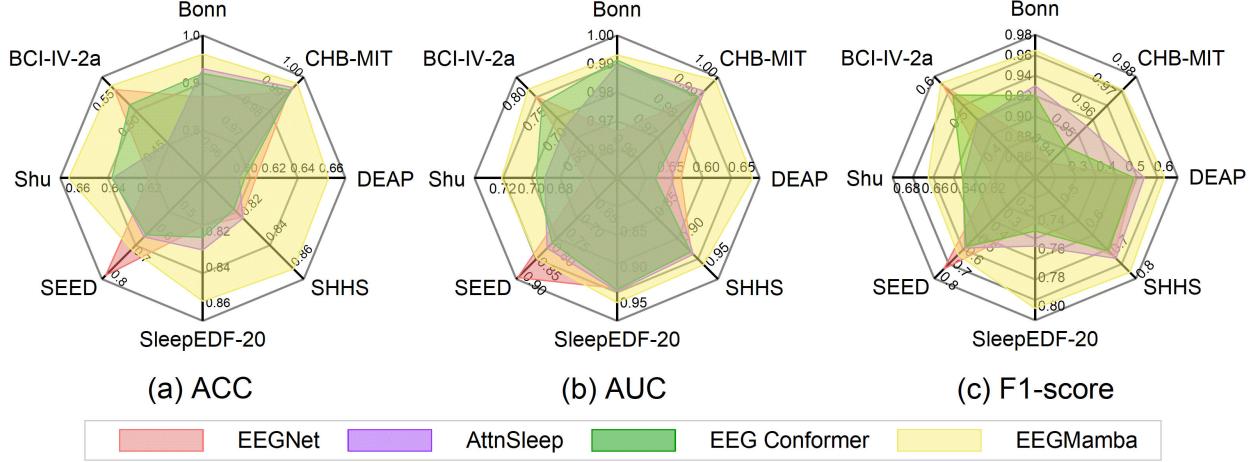


Figure 2: Performance comparison with other EEG classification models on different datasets.

We also discuss the memory-usage and inference speed of Single-task EEGMamba and Transformer-based models, especially when dealing with long sequences, as shown in Figure 3. Obviously, as the signal length increases, the memory usage of AttnSleep and EEG Conformer grows quadratically. When the signal length reaches 10000, the memory usage of Transformer-based models approaches the upper limit. In contrast, the memory usage of Single-task EEGMamba grows linearly with the signal length and can handle EEG signals of lengths exceeding 40000. In the comparison of inference speed, Single-task EEGMamba has no obvious advantage when the sequence length is less than 5000. However, as the sequence length increases, the inference speed of Transformer-based model decreases sharply, while that of Single-task EEGMamba decreases gently.

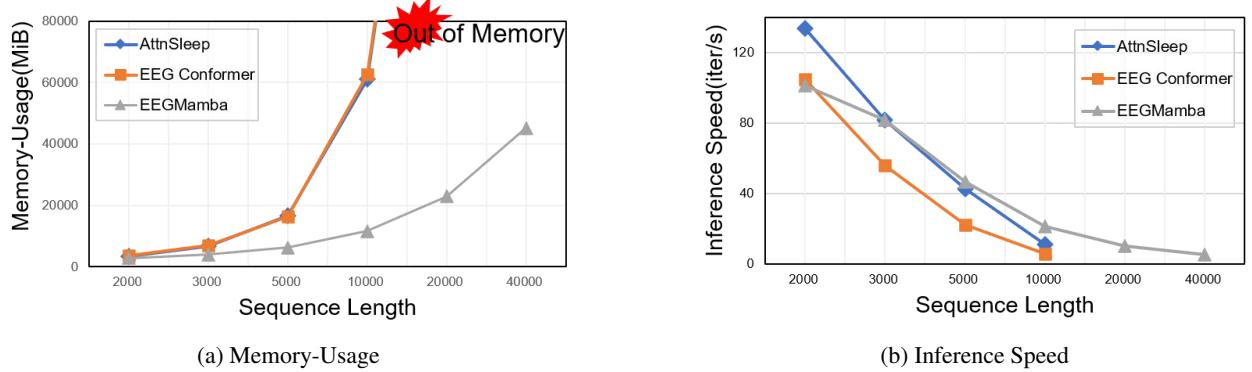


Figure 3: Memory-usage and inference speed of EEGMamba compared with Transformer-based models.

To summarize, compared with the previous classification networks, Single-task EEGMamba achieves better performance, lower memory usage and faster inference speed when dealing with long sequences.

4.2 EEGMamba for Multi-task EEG Classification

4.2.1 MoE in Multi-Task Classification

We build EEGMamba according to the structure described in Figure 1. Table 2 shows the accuracy of EEGMamba on different datasets compared with EEGNet Lawhern et al. [2018], Attnsleep Eldele et al. [2021], EEG Conformer Song et al. [2022], and single-task EEGMamba. It is worth noting that all classification networks, except for EEGMamba, are trained on a single dataset. Single datasets typically have consistency in data distribution, features, and labels, which allows the model to better adapt and optimize for the specific patterns and characteristics of that dataset, thereby improving accuracy. Even though, the performance of EEGMamba surpasses that of all classification networks except for Single-task EEGMamba. This reflects the ability of the proposed model to simultaneously process diverse EEG data.

Table 2: Accuracy of EEGMamba compared with other classification models on different datasets. Bold fonts indicate the highest accuracy, and red fonts indicate the second highest accuracy.

Classification Network	Universal Model	Epilepsy Detection		Sleep Stages Classification		Emotion Recognition		Motor Imagery	
		Bonn	CHB-MIT	SleepEDF-20	SHHS	DEAP	SEED	Shu	BCI-IV-2a
EEGNet	x	0.8700	0.9927	0.8269	0.8310	0.6172	0.8241	0.6301	0.5758
AttnSleep	x	0.9300	0.9947	0.8402	0.8320	0.6055	0.6562	0.6505	0.4695
EEG Conformer	x	0.9200	0.9936	0.8331	0.8247	0.6094	0.6469	0.6510	0.5453
Single-task EEGMamba	x	0.9600	0.9969	0.8689	0.8720	0.6680	0.7104	0.6751	0.5827
EEGMamba	✓	0.9300	0.9973	0.8500	0.8540	0.5234	0.7469	0.6430	0.4626

To obtain the corresponding results presented in Table 2, our EEGMamba only needs to be trained and set the input channels and the number of classes once. However, other classification networks need to be trained multiple times, requiring manual reconfiguration of data length, number of channels, and number of classes each time, which is very inconvenient.

Furthermore, we explore the role of MoE in EEGMamba through a series of experiments. In Table 3, we show the results of no-MoE, adding MoE in each Mamba block and adding MoE after all Mamba blocks (EEGMamba). It is evident that when performing multi-task classification, MoE can effectively enhance the model's capability. We also investigate the appropriate placement and number of MoEs. In the "MoE Each Mamba" configuration, we add one MoE for each Mamba block, resulting in a total of 8 MoE modules in the entire model. Although the model has become more complex, we have not observed significant performance improvements.

Table 3: The effect of different MoE usage on Multi-task EEGMamba.

Task	Dataset	no-MoE EEGMamba			MoE for Each Mamba			EEGMamba		
		ACC	AUC	F1	ACC	AUC	F1	ACC	AUC	F1
Epilepsy Detection	Bonn	0.8700	0.9850	0.8837	0.8900	0.9851	0.9045	0.9300	0.9917	0.9417
	CHB-MIT	0.9972	0.9971	0.9762	0.9978	0.9989	0.9810	0.9973	0.9958	0.9768
Sleep Stages Classification	SleepEDF-20	0.8348	0.9415	0.7580	0.8399	0.9471	0.7621	0.8500	0.9503	0.7784
	SHHS	0.8477	0.9364	0.7450	0.8483	0.9454	0.7350	0.8540	0.9512	0.7524
Emotion Recognition	DEAP	0.4961	0.4808	0.4488	0.5469	0.4935	0.5031	0.5234	0.5277	0.5127
	SEED	0.7427	0.8813	0.7410	0.7485	0.8828	0.7478	0.7469	0.8894	0.7448
Motor Imagery	Shu	0.6514	0.7181	0.6514	0.6735	0.7277	0.6729	0.6430	0.7104	0.6430
	BCI-IV-2a	0.4744	0.7146	0.4747	0.4380	0.6904	0.4392	0.4626	0.7051	0.4634

4.2.2 Contribution of Task-aware MoE

We explore the role of designed task-aware MoE in practical application. We calculate the probability that each expert will be activated in different tasks with and without task-aware MoE, as shown in Figure 4. When using task-aware MoE, the model shows an obvious expert selection preference for a given task, suggesting that each expert has task it is good at. However, when task-aware MoE is replaced by ordinary MoE, the step diagram is almost a straight line (Figure 4b), indicating that the difference in activation probability between experts is very subtle for the same task, which is contrary to our expectation of assigning different tasks to different experts for processing.

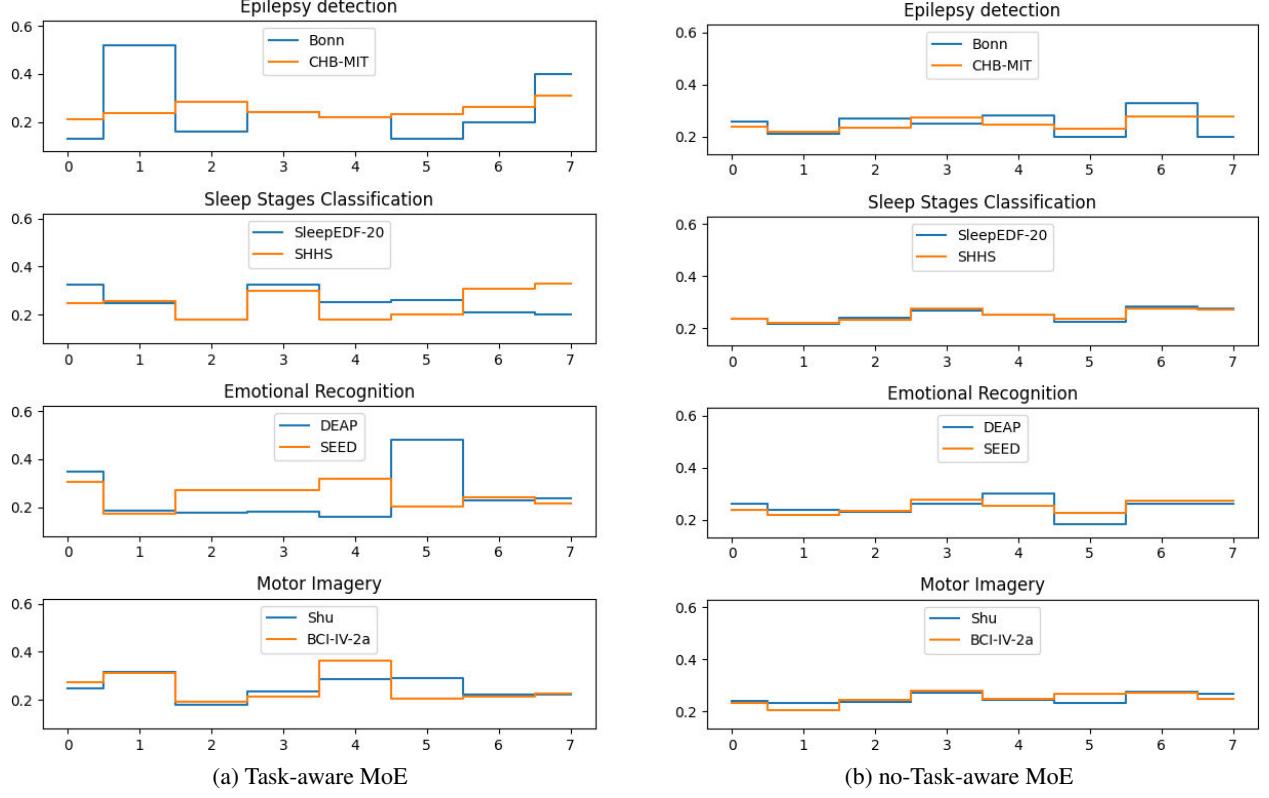


Figure 4: Expert activation probability with task-aware MoE and no-task-aware MoE.

4.3 Ablation Study

We compared the model performance using only task-aware gating, only universal expert, and both applied simultaneously, as shown in Table 4. Obviously, removing either of the task-aware gating and the universal expert will lead to a decrease in model performance compared to the task-aware MoE with universal expert. This may be due to the fact that the model after removing the modules cannot simultaneously capture the commonalities and differences between EEG data of different tasks. Therefore, the proposed task-aware MoE and universal experts can effectively enhance the model's performance.

Table 4: Module ablation study in task-aware MoE with universal expert.

Task	Dataset	only Universal Expert			only Task-aware Gating			Task-aware + Universal Expert		
		ACC	AUC	F1	ACC	AUC	F1	ACC	AUC	F1
Epilepsy Detection	Bonn	0.8700	0.9813	0.8806	0.9200	0.9770	0.9287	0.9300	0.9917	0.9417
	CHB-MIT	0.9960	0.9956	0.9641	0.9953	0.9924	0.9609	0.9973	0.9958	0.9768
Sleep Stages Classification	SleepEDF-20	0.8350	0.9417	0.7525	0.8447	0.9412	0.7721	0.8500	0.9503	0.7784
	SHHS	0.8491	0.9418	0.7279	0.8461	0.9579	0.7225	0.8540	0.9512	0.7524
Emotion Recognition	DEAP	0.5898	0.6028	0.5557	0.5547	0.5066	0.5164	0.5234	0.5277	0.5127
	SEED	0.7543	0.8924	0.7543	0.6393	0.8021	0.6390	0.7469	0.8894	0.7448
Motor Imagery	Shu	0.6555	0.7201	0.6555	0.6606	0.7165	0.6598	0.6430	0.7104	0.6430
	BCI-IV-2a	0.3927	0.6607	0.3915	0.4931	0.7248	0.4875	0.4626	0.7051	0.4634

5 Conclusion

In this paper, we propose EEGMamba, which utilizes ST-Adaptive module to adaptively extract features of EEG data with different lengths and channel numbers. We introduce bidirectional State Space Models SSM to achieve high accuracy and fast inference speed when processing long-term EEG datasets. We design a task-aware Mixture of Experts

(MoE) and an EEG universal expert, allowing the model to process multiple tasks simultaneously and better learn the commonalities among EEG signals from different tasks. We evaluate our model on eight publicly available EEG datasets across four tasks, and experimental results demonstrate the superior performance of our proposed model in multi-task classification scenarios.

References

- Afshin Shoeibi, Marjane Khodatars, Navid Ghassemi, Mahboobeh Jafari, Parisa Moridian, Roohallah Alizadehsani, Maryam Panahiazar, Fahime Khozeimeh, Assef Zare, Hossein Hosseini-Nejad, et al. Epileptic seizures detection using deep learning techniques: A review. *International journal of environmental research and public health*, 18(11): 5780, 2021.
- Mahboobeh Jafari, Afshin Shoeibi, Marjane Khodatars, Sara Bagherzadeh, Ahmad Shalbaf, David López García, Juan M Gorri, and U Rajendra Acharya. Emotion recognition in eeg signals using deep learning methods: A review. *Computers in Biology and Medicine*, page 107450, 2023.
- Hamdi Altaheri, Ghulam Muhammad, Mansour Alsulaiman, Syed Umar Amin, Ghadir Ali Altuwajri, Wadood Abdul, Mohamed A Bencherif, and Mohammed Faisal. Deep learning techniques for classification of electroencephalogram (eeg) motor imagery (mi) signals: A review. *Neural Computing and Applications*, 35(20):14681–14722, 2023.
- Tellakula Ramya Sri, Jahnavi Madala, Sai Lokesh Duddukuru, Rupasri Reddipalli, Phani Kumar Polasi, et al. A systematic review on deep learning models for sleep stage classification. In *2022 6th International Conference on Trends in Electronics and Informatics (ICOEI)*, pages 1505–1511. IEEE, 2022.
- Xun Chen, Chang Li, Aiping Liu, Martin J McKeown, Ruobing Qian, and Z Jane Wang. Toward open-world electroencephalogram decoding via deep learning: A comprehensive survey. *IEEE Signal Processing Magazine*, 39(2):117–134, 2022.
- Siavash Sakhavi, Cuntai Guan, and Shuicheng Yan. Learning temporal information for brain-computer interface using convolutional neural networks. *IEEE transactions on neural networks and learning systems*, 29(11):5619–5629, 2018.
- Punnawish Thuwajit, Phurin Rangpong, Phattarapong Sawangjai, Phairot Autthasan, Rattanaphon Chaisaen, Nannapas Banluesombatkul, Puttaranun Boonchit, Nattasate Tatsaringkansakul, Thapanun Sudhawiyangkul, and Theerawit Wilaiprasitporn. Eegwavenet: Multiscale cnn-based spatiotemporal feature extraction for eeg seizure detection. *IEEE transactions on industrial informatics*, 18(8):5547–5557, 2021.
- Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, 38(11):5391–5420, 2017.
- Yang Dai, Xiuli Li, Shanshan Liang, Lukang Wang, Qingtian Duan, Hui Yang, Chunqing Zhang, Xiaowei Chen, Longhui Li, Xingyi Li, et al. Multichannelsleepnet: A transformer-based model for automatic sleep stage classification with psg. *IEEE Journal of Biomedical and Health Informatics*, 2023.
- Jin Xie, Jie Zhang, Jiayao Sun, Zheng Ma, Liuni Qin, Guanglin Li, Huihui Zhou, and Yang Zhan. A transformer-based approach combining deep learning network and spatial-temporal information for raw eeg classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 30:2126–2136, 2022.
- Zhe Wang, Yongxiong Wang, Chuanfei Hu, Zhong Yin, and Yu Song. Transformers for eeg-based emotion recognition: A hierarchical spatial information learning model. *IEEE Sensors Journal*, 22(5):4359–4368, 2022.
- Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.
- Alison O’Shea, Gordon Lightbody, Geraldine Boylan, and Andriy Temko. Neonatal seizure detection from raw multi-channel eeg using a fully convolutional architecture. *Neural Networks*, 123:12–25, 2020.
- Huy Phan, Kaare Mikkelsen, Oliver Y Chén, Philipp Koch, Alfred Mertins, and Maarten De Vos. Sleeptransformer: Automatic sleep staging with interpretability and uncertainty quantification. *IEEE Transactions on Biomedical Engineering*, 69(8):2456–2467, 2022.
- Mona Algarni, Faisal Saeed, Tawfik Al-Hadhrami, Fahad Ghabban, and Mohammed Al-Sarem. Deep learning-based approach for emotion recognition using electroencephalography (eeg) signals using bi-directional long short-term memory (bi-lstm). *Sensors*, 22(8):2976, 2022.
- Phairot Autthasan, Rattanaphon Chaisaen, Thapanun Sudhawiyangkul, Phurin Rangpong, Suktipol Kiatthaveephong, Nat Dilokthanakul, Gun Bhakdisongkhram, Huy Phan, Cuntai Guan, and Theerawit Wilaiprasitporn. Min2net: End-to-end multi-task learning for subject-independent motor imagery eeg classification. *IEEE Transactions on Biomedical Engineering*, 69(6):2105–2118, 2021.
- Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces. *Journal of neural engineering*, 15(5):056013, 2018.

- Hangyu Zhu, Laishuan Wang, Ning Shen, Yonglin Wu, Shu Feng, Yan Xu, Chen Chen, and Wei Chen. Ms-hnn: Multi-scale hierarchical neural network with squeeze and excitation block for neonatal sleep staging using a single-channel eeg. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:2195–2204, 2023a.
- Hangyu Zhu, Wei Zhou, Cong Fu, Yonglin Wu, Ning Shen, Feng Shu, Huan Yu, Wei Chen, and Chen Chen. Masksleepnet: A cross-modality adaptation neural network for heterogeneous signals processing in sleep staging. *IEEE Journal of Biomedical and Health Informatics*, 27(5):2353–2364, 2023b.
- Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.
- Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2016.
- Fuzhao Xue, Zian Zheng, Yao Fu, Jinjie Ni, Zangwei Zheng, Wangchunshu Zhou, and Yang You. Openmoe: An early effort on open mixture-of-experts language models. *arXiv preprint arXiv:2402.01739*, 2024.
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.
- Yunhao Gou, Zhili Liu, Kai Chen, Lanqing Hong, Hang Xu, Aoxue Li, Dit-Yan Yeung, James T Kwok, and Yu Zhang. Mixture of cluster-conditional lora experts for vision-language instruction tuning. *arXiv preprint arXiv:2312.12379*, 2023.
- Ralph G Andrzejak, Klaus Lehnertz, Florian Mormann, Christoph Rieke, Peter David, and Christian E Elger. Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Physical Review E*, 64(6):061907, 2001.
- Ali Hossam Shoeb. *Application of machine learning to epileptic seizure onset detection and treatment*. PhD thesis, Massachusetts Institute of Technology, 2009.
- Bob Kemp, Aeilko H Zwinderman, Bert Tuk, Hilbert AC Kamphuisen, and Josefien JL Oberye. Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the eeg. *IEEE Transactions on Biomedical Engineering*, 47(9):1185–1194, 2000.
- Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220, 2000.
- Sander Koelstra, Christian Muhl, Mohammad Soleymani, Jong-Seok Lee, Ashkan Yazdani, Touradj Ebrahimi, Thierry Pun, Anton Nijholt, and Ioannis Patras. Deap: A database for emotion analysis; using physiological signals. *IEEE transactions on affective computing*, 3(1):18–31, 2011.
- Ruo-Nan Duan, Jia-Yi Zhu, and Bao-Liang Lu. Differential entropy feature for eeg-based emotion classification. In *2013 6th international IEEE/EMBS conference on neural engineering (NER)*, pages 81–84. IEEE, 2013.
- Jun Ma, Banghua Yang, Wenzheng Qiu, Yunzhe Li, Shouwei Gao, and Xinxing Xia. A large eeg dataset for studying cross-session variability in motor imagery brain-computer interface. *Scientific Data*, 9(1):531, 2022.
- Clemens Brunner, Robert Leeb, Gernot Müller-Putz, Alois Schlögl, and Gert Pfurtscheller. Bci competition 2008–graz data set a. *Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology*, 16:1–6, 2008.
- Pedro Fonseca, Niek Den Teuling, Xi Long, and Ronald M Aarts. Cardiorespiratory sleep stage detection using conditional random fields. *IEEE journal of biomedical and health informatics*, 21(4):956–966, 2016.
- Emadeldeen Eldele, Zhenghua Chen, Chengyu Liu, Min Wu, Chee-Keong Kwoh, Xiaoli Li, and Cuntai Guan. An attention-based deep learning approach for sleep stage classification with single-channel eeg. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 29:809–818, 2021.
- Yonghao Song, Qingqing Zheng, Bingchuan Liu, and Xiaorong Gao. Eeg conformer: Convolutional transformer for eeg decoding and visualization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:710–719, 2022.
- Akara Supratak, Hao Dong, Chao Wu, and Yike Guo. Deepsleepnet: A model for automatic sleep stage scoring based on raw single-channel eeg. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(11):1998–2008, 2017.

- Zhenghua Chen, Min Wu, Wei Cui, Chengyu Liu, and Xiaoli Li. An attention based cnn-lstm approach for sleep-wake detection with heterogeneous sensors. *IEEE Journal of Biomedical and Health Informatics*, 25(9):3270–3277, 2020.
- Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021.
- Jimmy TH Smith, Andrew Warrington, and Scott W Linderman. Simplified state space layers for sequence modeling. *arXiv preprint arXiv:2208.04933*, 2022.
- Daniel Y Fu, Tri Dao, Khaled K Saab, Armin W Thomas, Atri Rudra, and Christopher Ré. Hungry hungry hippos: Towards language modeling with state space models. *arXiv preprint arXiv:2212.14052*, 2022.
- Suparerk Janjarasjitt. Epileptic seizure classifications of single-channel scalp eeg data using wavelet-based features and svm. *Medical & biological engineering & computing*, 55(10):1743–1761, 2017.
- Edward A Wolpert. A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects. *Archives of General Psychiatry*, 20(2):246–247, 1969.
- Phan Huy, Fernando Andreotti, Navin Cooray, Oliver Y Chen, and Maarten De Vos. Seqsleepnet: End-to-end hierarchical recurrent neural network for sequence-to-sequence automatic sleep staging. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 27(3):400–410, 2019.
- Muhammad Khateeb, Syed Muhammad Anwar, and Majdi Alnowami. Multi-domain feature fusion for emotion classification using deap dataset. *IEEE Access*, 9:12134–12142, 2021.
- Barret Zoph, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer, and William Fedus. St-moe: Designing stable and transferable sparse expert models. *arXiv preprint arXiv:2202.08906*, 2022.

A Appendix

A.1 Related Works

A.1.1 EEG Classification

The development of deep learning has greatly advanced EEG classification tasks. CNNs are a classic type of neural network with mature applications in EEG classification. Schirrmeister et al. [2017] proposed a shallow convolutional network with both spatiotemporal convolutional layers to decode task-related information from raw EEG signals. Similarly, Lawhern et al. [2018] introduced EEGNet, a classic EEG classification network based on depthwise separable convolution, which has demonstrated stable and robust performance in various EEG classification tasks. Recurrent Neural Networks (RNNs) are proposed to capture temporal dependencies in time-series EEG signals. Supratak et al. [2017] used the RNN architecture for sleep stage classification. Chen et al. [2020] used CNN and Long Short Term Memory (LSTM) networks for sleep stage classification. EEG classification networks based on Transformers have also made significant progress. Eldele et al. [2021] introduced attention mechanisms into EEG classification networks for classifying sleep stages. Song et al. [2022] proposed EEG Conformer, a EEG classification network based on spatiotemporal convolution and Transformers. EEG Conformer effectively extracts local and global features from EEG signals, and it performs well in tasks such as motor imagery and emotion recognition.

A.1.2 State Space Model

A state-space model is a mathematical model that represents a physical system as a set of input, output, and state variables related by a first-order differential equation. Gu et al. [2021] proposed the Structured State-Space Sequence Model (S4) to model long-term dependencies. Smith et al. [2022] introduced a new S5 layer by incorporating Multiple Input Multiple Output (MIMO) SSM and efficient parallel scanning within the S4 layer. Fu et al. [2022] designed a new SSM layer, H3, which further narrowed the performance gap between SSM and Transformers. Recently, Gu et al. [2023] proposed a data-dependent SSM structure and built a universal language model backbone network: Mamba. Its selective mechanism and hardware-aware design allow it to maintain computational efficiency and excellent performance while scaling to billions of parameters.

A.1.3 Mixture of Experts

The Mixture of Experts model was first introduced by Jacobs et al. [1991], which controls a system composed of different networks called experts through a supervisory program, with each expert responsible for handling a specific subset of training samples. Shazeer et al. [2016] introduced the concept of sparsity into MoE and applied it to LSTM models for translation tasks. With the development of large language models, Fedus et al. [2022] extensively investigated the stability issues of MoE models during training and fine-tuning processes, and built a MoE model with 16 trillion parameters and 2048 experts. Recently, Xue et al. [2022] proposed OpenMOE, which further explores the details of MoE using the power of the open-source community, thereby promoting the development of MoE.

A.2 Dataset

A.2.1 Bonn

The Bonn dataset is composed of EEG data from 5 healthy individuals and 5 patients with epilepsy, totaling 5 subsets, which are F, S, N, Z, and O. The Bonn dataset is a single-channel dataset, where each subset contains 100 data segments. The duration of each data segment is 23.6 seconds with a sampling frequency of 173.61Hz. During the data cutting process, noise signals such as myogenic and ocular artifacts have been removed. The data Z and O are scalp EEGs, collected from 5 healthy individuals, forming the control group. The segments in Z are EEGs when the subjects have their eyes open, and the segments in O are EEGs when the subjects have their eyes closed. The data N, F, and S are intracranial EEGs, collected from 5 patients who have been diagnosed preoperatively. N and F are collected during the interictal phase of epilepsy, and S is collected during the ictal phase. To facilitate model processing, we have truncated the length of this dataset to 4,096 signal points.

A.2.2 CHB-MIT

The CHB-MIT scalp EEG database is collected by the Children's Hospital Boston, which contains 24 cases of 23 patients with intractable seizures. The first 23 cases are from 22 patients (17 females, aged 1.5-19 years; 5 males, aged 3-22 years). For the last case, there is no clear gender or age record. The Children's Hospital Boston evaluated the potential conditions for surgical intervention in all epilepsy patients after discontinuing medication for a period of time, and monitored the patients for several days. The original EEG record was obtained using 256 Hz sampling rate with 16-bit resolution from electrodes placed according to the international 10-20 EEG electrode positions and nomenclature [Janjarasjitt 2017]. Given that the number of available channels varies among different patients, we select 23 common channels and discarded data from less than 23 channels. Due to the varying duration of the original data ranging from tens of minutes to several hours, we have truncated it into 4-second segments for easy classification.

A.2.3 SleepEDF-20

SleepEDF-20 includes Polysomnography (PSG) records from each subject for two consecutive days and nights. The recording of subject 13 on the second night was lost due to a failing cassette or laserdisc. Sleep experts use R&K rules [Wolpert 1969] to visually determine signal characteristics and label each 30 second period in the dataset as one of eight stages W, N1, N2, N3, N4, REM, MOVEMENT, UNKNOWN. Similar to previous work Huy et al. [2019], N3 and N4 were merged into N3. In addition, the stages of "MOVEMENT" and "UNKNOWN" have also been removed. In our experiment, Fpz-Cz EEG with a sampling rate of 100Hz was adopted for sleep staging.

A.2.4 SHHS

Sleep Heart Health Study (SHHS) is a multi-center cohort study on the cardiovascular and other consequences associated with sleep apnea. The research subjects suffer from various diseases, including lung disease, cardiovascular disease, and coronary heart disease. To reduce the impact of these diseases, we referred to the research method of Fonseca et al. [2016] and selected subjects who were considered to have regular sleep patterns (such as those with apnea hypopnea index (AHI) less than 5). Finally, we select 329 for the experiment out of 6441 participants. It is worth noting that we chose the C4-A1 channel with a sampling rate of 125 Hz.

A.2.5 DEAP

In the DEAP dataset, movies are used as emotional inducers in experiments. This dataset contains data from over 32 participants aged between 19 and 37, half of whom are females. Participants sit one meter away from the screen. The device records EEG signals at a sampling rate of 512Hz. 40 selected music video clips were used to trigger emotions. At the end of each video, participants were asked to evaluate their level of arousal, valence, preference, and dominance. The self-assessment scale ranges from 1 to 9. The scores of the subjects are divided into two categories (low or high) based on a stable threshold of 4.5. During the preprocessing process, the EEG signal is downsampled to 128Hz and a bandpass filter with a cutoff frequency of 4-45Hz is applied. In this paper, we use the same channel selection as Khateeb et al. [2021], which includes four electrodes: FP1, FP2, F3, and C4.

A.2.6 SEED

The SEED dataset collects EEG data from 15 participants while watching emotional movies. It contains a total of 45 experiments. The EEG data is collected by 62 channels based on the international 10-20 system and a sampling rate of 1000Hz. During the preprocessing process, the data is downsampled to 200Hz and subjected to a bandpass filter ranging from 0 to 75Hz. The extraction of EEG sections was based on the duration of each movie. Within each subject's data

file, there are 16 arrays, with 15 of these arrays containing 15 preprocessed segments of EEG data from the experiment. The label array includes corresponding emotional labels, where 1 for positive, 2 for negative, and 3 for neutral emotions.

A.2.7 Shu

The motor imagery dataset experiment consists of three phases. The first phase (0-2 seconds) is the resting preparation period, during which subjects can rest, perform minor physical activities, and blink. The second phase (2-4 seconds) is the cue phase, where an animation of left or right hand movement appears on the monitor, indicating the upcoming task. The third phase (4-8 seconds) is the MI (Motor Imagery) phase, during which subjects perform the hand movement MI task as prompted, and EEG signals are recorded. Each session consists of 100 trials, with five sessions conducted for each subject every 2 to 3 days, resulting in a total of 500 trials per subject.

A.2.8 BCI-IV-2a

The BCI-IV-2a dataset includes EEG signals obtained from trials involving 9 subjects. This experiment includes four different motor imagery tasks: left hand, right hand, foot, and tongue. Each participant participated in two training sessions, with six sessions per session. In each run, there were 48 trials, a total of 288 trials (12 trials per MI task, a total of 72 trials per task). A set of 25 Ag/AgCl electrodes were used in the experiment, of which 22 were dedicated to recording EEG signals, while the remaining three electrodes recorded eye movement signals (not used in our experiment). All recorded signals are processed through a bandpass filter of 0.5 to 100Hz and a 50Hz notch filter. The sampling frequency is set to 250Hz.

A.3 Experimental Related Supplements

A.3.1 Load Balance and Model stability in MoE

Training an MoE typically encounters two issues: (1) Load imbalance: the gating network tends to select only a few experts. (2) Training instability: excessively large gating values for a few experts lead to an unstable training process. To address these issues, we incorporate balance loss L_b Shazeer et al. [2016] and router z-loss L_z Zoph et al. [2022] as auxiliary losses for the model to mitigate load imbalance and training instability, as shown in Equation (11), where B represents the batch size.

$$L_b = \frac{Std(e(t_{cls}^*))}{Mean(e(t_{cls}^*))} \quad (11)$$

$$L_z = \frac{1}{B} \sum_{i=1}^B (\log(\exp(t_{cls}^*)))^2$$

$$L_{aux} = L_b + L_z$$

A.3.2 Visualization of Features Extracted by Bidirectional Mamba

Figure 5 shows t-distributed stochastic neighbor embedding (t-SNE) plots of features extracted by Single-task EEG-Mamba from different datasets. The plot exhibits distinct distances between features of different classes and small distances within the same class, indicating the successful extraction of features from different classes by EEGMamba. This may indicate its comprehensive performance superiority across different datasets.

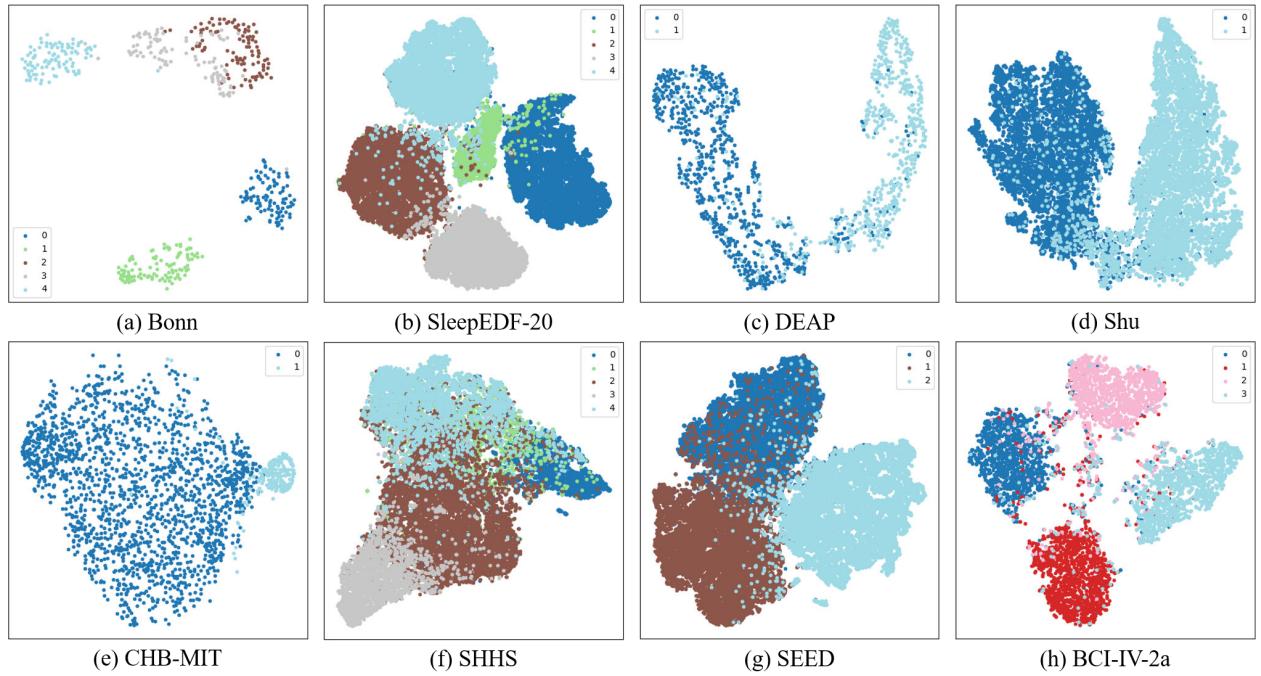


Figure 5: Visualization results of feature extracted by EEGMamba on different datasets.

A.3.3 Bidirectional Mamba Ablation Study

We analyze the role of each module in EEGMamba through a series of ablation analysis, as shown in Table 5. When using Mamba, the accuracy decreases on datasets except CHB-MIT and SleepEDF-20. This indicates that bidirectional modeling can better capture the dependency relationship between EEG sequences. We also attempt to combine single-directional causal convolution with bidirectional SSM, and the results are similar to single-directional modeling. Overall, the combination of bidirectional SSM and bidirectional causal convolution leads to better performance.

Table 5: Module ablation study in bidirectional Mamba. I represents the combination of single-directional causal convolution and single-directional SSM. II represents the combination of single-directional causal convolution and bidirectional SSM. III represents the combination of bidirectional causal convolution and bidirectional SSM used by EEGMamba.

Task	Dataset	I			II			III		
		ACC	AUC	F1	ACC	AUC	F1	ACC	AUC	F1
Epilepsy Detection	Bonn	93.00%	0.9914	0.9345	95.00%	0.9923	0.9564	96.00%	0.9931	0.9644
	CHB-MIT	99.70%	0.9987	0.9748	99.68%	0.9985	0.9728	99.69%	0.9986	0.9734
Sleep Stages Classification	SleepEDF-20	87.07%	0.9753	0.8141	86.56%	0.9742	0.8060	86.89%	0.9742	0.8117
	SHHS	86.96%	0.9696	0.7698	86.85%	0.9700	0.7720	87.20%	0.9714	0.7753
Emotion Recognition	DEAP	66.02%	0.6751	0.6448	66.41%	0.6771	0.6426	66.80%	0.6897	0.6529
	SEED	70.42%	0.8728	0.7026	70.57%	0.8729	0.7065	71.04%	0.8788	0.7108
Motor Imagery	Shu	67.14%	0.7295	0.6714	66.85%	0.7339	0.6682	67.51%	0.7310	0.6749
	BCI-IV-2a	56.79%	0.8106	0.5678	58.27%	0.8155	0.5827	58.27%	0.8199	0.5802

A.3.4 The Original Results of Single-task Mamba Comparison Experiment

Table 6 records the original results of the Single-task EEGMamba comparison experiment, and its visualization results are shown in Figure 2.

Table 6: The original results of Single-task Mamba comparison experiment.

Task	Dataset	EEGNet			AttnSleep			EEG Conformer			EEGMamba		
		ACC	AUC	F1	ACC	AUC	F1	ACC	AUC	F1	ACC	AUC	F1
Epilepsy Detection	Bonn	0.8700	0.9662	0.8729	0.9300	0.9896	0.9301	0.9200	0.9913	0.9205	0.9600	0.9931	0.9644
	CHB-MIT	0.9927	0.9909	0.9373	0.9947	0.9931	0.9554	0.9936	0.9903	0.9451	0.9969	0.9986	0.9734
Sleep Stages Classification	SleepEDF20	0.8269	0.9545	0.7518	0.8402	0.9585	0.7680	0.8331	0.9561	0.7575	0.8689	0.9742	0.8117
	SHHS	0.8310	0.9474	0.7130	0.8320	0.9510	0.7201	0.8247	0.9454	0.6907	0.8720	0.9714	0.7753
Emotion Recognition	DEAP	0.6172	0.5898	0.5483	0.6055	0.5757	0.5825	0.6094	0.5533	0.5455	0.6680	0.6897	0.6529
	SEED	0.8241	0.9444	0.8241	0.6562	0.8429	0.6584	0.6469	0.8245	0.6479	0.7104	0.8788	0.7108
Motor Imagery	Shu	0.6301	0.6725	0.6301	0.6505	0.7008	0.6505	0.6510	0.7067	0.6509	0.6751	0.7310	0.6749
	BCI-IV-2a	0.5758	0.8026	0.5751	0.4695	0.7283	0.4698	0.5453	0.7855	0.5452	0.5827	0.8199	0.5802

A.4 Limitations

Although the current experimental results show that EEGMamba can be well applied to EEG multi-task classification, it still has some limitations. On the one hand, this paper only covers four kinds of EEG tasks to verify the performance of EEGMamba, which is only a small part of the tasks that EEG can accomplish. Therefore, EEGMamba is still far from a universal EEG classification model. On the other hand, it should be extended to other one-dimensional time signals besides EEG to prove the universality of the model in one-dimensional time signals.

Exploring Large-Scale Language Models to Evaluate EEG-Based Multimodal Data for Mental Health

Yongquan Hu

yongquan.hu@unsw.edu.au
University of New South Wales
Sydney, NSW, Australia

Hong Jia

h.jia.cam@gmail.com
University of Melbourne
Melbourne, VIC, Australia

Shuning Zhang

zsn23@mails.tsinghua.edu.cn
Tsinghua University
Beijing, China

Ting Dang

ting.dang@unimelb.edu.au
University of Melbourne
Melbourne, VIC, Australia

Flora D. Salim

flora.salim@unsw.edu.au
University of New South Wales
Sydney, NSW, Australia

Wen Hu

wen.hu@unsw.edu.au
University of New South Wales
Sydney, NSW, Australia

Aaron J. Quigley

aquigley@acm.org
CSIRO's Data61
Sydney, NSW, Australia

Abstract

Integrating physiological signals such as electroencephalogram (EEG), with other data such as interview audio, may offer valuable multimodal insights into psychological states or neurological disorders. Recent advancements with Large Language Models (LLMs) position them as prospective “health agents” for mental health assessment. However, current research predominantly focus on single data modalities, presenting an opportunity to advance understanding through multimodal data. Our study aims to advance this approach by investigating multimodal data using LLMs for mental health assessment, specifically through zero-shot and few-shot prompting. Three datasets are adopted for depression and emotion classifications incorporating EEG, facial expressions, and audio (text). The results indicate that multimodal information confers substantial advantages over single modality approaches in mental health assessment. Notably, integrating EEG alongside commonly used LLM modalities such as audio and images demonstrates promising potential. Moreover, our findings reveal that 1-shot learning offers greater benefits compared to zero-shot learning methods.

CCS Concepts

- Human-centered computing → Ubiquitous and mobile computing;
- Applied computing → Life and medical sciences.

Keywords

Mental Health, EEG, Large Language Model, Prompt Engineering.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

UbiComp Companion '24, October 5–9, 2024, Melbourne, VIC, Australia

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-1058-2/24/10
<https://doi.org/10.1145/3675094.3678494>

ACM Reference Format:

Yongquan Hu, Shuning Zhang, Ting Dang, Hong Jia, Flora D. Salim, Wen Hu, and Aaron J. Quigley. 2024. Exploring Large-Scale Language Models to Evaluate EEG-Based Multimodal Data for Mental Health. In *Companion of the 2024 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp Companion '24), October 5–9, 2024, Melbourne, VIC, Australia*. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3675094.3678494>

1 Introduction

Mental health, as defined by the World Health Organization (WHO), is a state of well-being where individuals can realise their potential, handle normal life stresses, work productively, and contribute to their communities [31]. Mental health issues are increasingly impacting the global economy [11], with conditions such as depression and anxiety estimated to cost trillions of dollars in lost productivity annually [8].

The accurate measurement and classification of such health conditions requires psychological evaluation which can include the recording of various indicators. Commonly, many physiological signals, such as Electroencephalogram (EEG) [12], Heart Rate Variability (HRV) [15], and Electrodermal Activity (EDA) [14], are integral for mental health assessments due to their reliability and difficulty to mask, ensuring more accurate identification [18, 40]. These signals are readily captured by widely available sensors [2, 10, 35].

In addition to capturing data, advancements in Artificial Intelligence (AI) technology have led researchers to develop various algorithms (e.g., machine learning) for the timely and accurately detection [1], modeling [43] and inference [29] of health conditions based on physiological signals. Recently, the capabilities of Large-scale Language Models (LLMs) have introduced a new paradigm for prediction and assessment in mental health [24, 42, 44]. LLMs offer several advantages, including enhanced multimodal data processing for improved assessment accuracy [25], interactive communication methods like human-in-the-loop to create more configurable health agents [4], and the potential for fine-tuning domain-specific purpose based on general models to reduce costs [39, 44]. However, most work using LLMs to detect mental health focuses on tasks of

single modality data such as Mental-LLM [44] and EEG-GPT [21], and the exploration of LLMs in evaluating multimodal sensing data for mental health remains limited. Moreover, existing multimodal LLMs have been developed primarily using audio and video modalities. They may lack the capabilities in handling other types of data, such as EEG and other physiological signals which play a crucial role [12] in mental health assessment. Among various physiological signals, EEG is particularly valuable, providing high-frequency data that accurately assesses conditions such as depression, mood, and stress levels [5]. Therefore, understanding how these LLMs process EEG data and how to effectively combine EEG with existing modalities remains an open question.

This paper introduces MultiEEG-GPT, a method for assessing mental health using multimodalities, especially with EEG, i.e., EEG and facial expression or audio. The latest GPT-4o API¹ is adopted for processing multimodalities to recognize the health conditions. Unlike its predecessors such as GPT-4 and GPT-4v, which require separate interface calls, GPT-4o integrates multimodal data processing into a single interface, enhancing the development of this method [30]. This work aims to understand the capabilities of multimodal LLMs in categorising various mental health conditions. This work seeks to compare their ability to model different modalities and EEG and design optimal prompt engineering to facilitate reliable prediction.

The contributions of this paper include: i) the prompt engineering design using both zero-shot and few-shot approaches to examine the predictive capability of MultiEEG-GPT using multimodalities in recognizing different health conditions; ii) experiments across three different databases to validate the effectiveness of MultiEEG-GPT. iii) an in-depth analysis to understand how multimodalities enhance health condition predictions compared to single modalities. We aim to open up further developments, such as health-supportive social robots [4, 19, 23], within the context of ubiquitous computing, human-computer interaction, and affective computing.

2 Related Work

EEG-based physiological signal analysis has long been essential for monitoring mental health, evolving alongside AI advancements. Initially focused on traditional machine algorithms like k-nearest Neighbor (k-NN) and Support Vector Machine (SVM) for EEG data, Hou et al. demonstrated the potential of EEG for stress level recognition, with the accuracy of 67.07% [17]. Later, the field has shifted towards integrating deep learning and multimodal data. Zhongjie et al. developed a fusion algorithm levering deep neural networks that combines Convolutional Neural Networks (CNNs) and Bidirectional Long Short-Term Memory (BiLSTM) networks for emotion classification, markedly demonstrating the impressive accuracy in valence and arousal classifications to $93.20 \pm 2.55\%$ and $93.18 \pm 2.71\%$, respectively [26].

Recently, the advent of general LLMs capable of processing multimodal data has further pivoted the focus towards using LLMs for evaluating mental health data, anticipating their role as future evaluation agents. For example, Xuhai et al. tested various LLMs, including GPT-3.5 and GPT-4, across multiple datasets using methods like zero-shot and few-shot prompting [44]. Jonathan et al.

introduced EEG-GPT, using GPT models to classify and interpret EEG data [21]. However, these studies still focus on single modality, such as text or EEG. Given various modalities can provide rich and complementary information to infer health conditions, it is proposed to consider different modalities in the automatic systems as well, especially with EEG in many mental health applications. However, research on LLMs for multimodal data with EEG is still limited for mental health prediction. Our proposed MultiEEG-GPT pioneers the work in examining multimodal data including EEG to infer health conditions, aiming to bridge this gap by enhancing the processing of multimodal signals, with a particular focus on EEG data.

3 Methodology

3.1 Dataset Selection

Various mental health dataset existed, of which numerous contained EEG modality. Applying the criteria that the dataset need to contain at least EEG modality, we selected 3 most commonly used datasets: (1) MODMA [5] was developed by Hanshu et al., and this multimodal dataset is designed for analyzing depression disorders and includes oral records (audio) of both patients and controls, and EEG data (convertible to images) from these groups. This dataset has binary labels of whether the participant was diagnosed with Major Depressive Disorder (MDD). (2) PME4 [7] is a multimodal emotion dataset featuring four modalities: audio, video (not publicly available), EEG, and electromyography (EMG) [7]. It was collected from 11 acting students (five female and six male) who provided informed consent. This dataset focuses on identifying seven emotions: anger, fear, disgust, sadness, happiness, surprise, and a neutral state; (3) LUMED-2 [9] was collected by Loughborough University and Hacettepe University, and it was designed to analyze facial expression, EEG, and galvanic skin response (GSR) data to recognize and classify three categories of human emotions (neutral, happy, sad) under various stimuli , advancing the understanding in affective computing.

For MODMA and PME4, we used audio and EEG modalities, while for LUMED-2, we used facial expression and EEG modalities. We chose audio and facial expression features because they were the among the most prevalent modalities in mental health analysis [28, 37]. Besides, the focus of this paper was to explore the possibility of GPT to analyze multimodal data, particularly with the important EEG modality [16]. Thus, we did not include the physiological modalities (e.g., GSR, Resp).

3.2 Prompt Design

For our MultiEEG-GPT method, we use prompt engineering strategies (including zero-shot prompting and few-shot prompting) for prediction tasks on multiple datasets. These prompts are model-agnostic, and we present the details of language models and settings employed for our experiment in the next section.

For the prompting strategies, we built upon the design in [44] and [45]. We have designed the prompt to account for different modalities and incorporated flexibility in altering the number of modalities for evaluation. Additionally, we have verified and compared few-shot and zero-shot prompts for evaluation.

¹<https://openai.com/index/hello-GPT-4o/>, accessed on June 11, 2024

Zero-shot prompting. As shown in Table 1, the zero-shot prompting strategy consists of a role-play prompt, a specific task description, and an additional rule to avoid unnecessary output and restricted models to focus on the current task. The role-play prompt aims to inform the LLMs of the general task, while the specific task description provides the information for different modalities. Such description also provides the flexibility in adding or deleting modalities. Therefore, the final prompt for the model consisted of: {role-play prompt} + {task specification} + {rules}.

Few-shot prompting. The few-shot prompt added the few-shot samples after the same zero-shot prompt template. Specifically, we include the task-specific prompt followed the zero-shot prompt, but providing the correct class labels instead of offering different candidate class labels for prediction, which is similar to Xuhai et al's setting [44].

Table 1: The zero-shot and few-shot prompting strategies. <MOD1>, <MOD2> and <MOD3> as placeholders denote three different modalities. XXX is the description of collection and visualization process. <SYM> as a placeholder denotes the symptom to be diagnosed. For example, for depression analysis <SYM> should be replaced with depression. The example is for mental health diagnosis with three classes. The label description “0 denotes XXX” of the classes could be added or removed to accommodate for more or less classes.

Role-play prompt	Imagine you are a mental health expert expert at analyzing the emotion and mental health status.
Task specification	The below is <MOD1>, <MOD2> and <MOD3> data. <MOD1> data is collected through XXX and visualized in XXX form. <MOD2> data is collected through XXX and visualized in XXX form. <MOD3> data is collected through XXX and visualized in XXX form. Analyze the <SYM> status of the person. 0 denotes XXX, 1 denotes XXX, 2 denotes XXX.
Rules	[Rule]: Do not output other text.

4 Experiment

4.1 Settings

4.1.1 Dataset Settings. As all the datasets used standard 10-20 electrode layout, we set the electrodes following this layout. MNE library is used for processing EEG signal. We processed the datasets using the raw data instead of their pre-processed data (e.g., PME4) because the pre-processed data only contained features instead of the original signals, which were infeasible for plotting topology map. We used a bandpass filter (low-frequency cutoff 0.1Hz, high-frequency cutoff 45Hz, Hamming Window) [34] with firwin window design. Afterward, the filtered data were re-referenced to an average reference [34]. Since the elicitation presented with different length for different datasets, we chose 530s, 5s and 1.65-4.15s for LUMED-2, PME4 and MODMA datasets respectively, to account for randomly set elicitation time , with 10 equidistant sampled

timestamps to create topology maps. For the facial expression, we chose the middle frame of the video (e.g., if the video's length is 10s, we chose the frame at exactly the 5s timestamp) or the image. For the audio, because GPT-4o² have not yet released the audio input support, we used both the audio features and the text as inputs. For the audio features, we used librosa library to extract the features from the audio and represent these features in text format (which is similar to EEG-GPT's representation [21]), which includes MFCCs, Mel Spectrogram, Chroma STFT, etc. For the text, we transcribed the audio using automatic speech recognition (ASR) systems. We chose the open-sourced vosk library³ with vosk-model-cn-0.15 (Chinese version) or vosk-model-en-0.22 (English version) according to the need. These models were the largest and most advanced ASR systems in the vosk library, which ensured the accuracy of recognition and was used in health care tasks [13, 32].

4.1.2 Model Settings. For all datasets and all tasks, we transformed the tasks into multi-class classification problems as in previous work [21, 44]. For MODMA, the binary class labels are ‘MDD’ or ‘healthy’. For PME4, we followed the labels in the original datasets to classify the emotion into seven classes: anger, fear, disgust, sadness, happiness, surprise and neutral. For LUMED-2, we set the 3-class labels as in the original paper, which included neutral, happy and sadness.

Previous work showed that GPT-4 generally performed better than GPT-3.5 [44]. Given that GPT-4o is the most recent series of GPT-4 that naturally supports multimodal capabilities, we adopted GPT-4o as the tested LLMs. Specifically, we used “GPT-4o-2024-05-13”⁴ as the targeted model through OpenAI Azure’s API⁵. For the few-shot experiment, we tested the 1-shot learning scenario to examine the capability of multi-model LLMs with limited information provided. In each repeated trial, we randomly selected one sample from the corresponding dataset to act as the 1-shot sample. This strategy mitigates the bias of selecting samples. For all zero-shot and few-shot experiments, we tested across each dataset (for the few-shot experiment, we excluded that selected sample) for 5 times and reported the average accuracy and the standard deviation.

We use the image updating module of GPT-4o. However, we use no other any additional techniques (e.g., Chain-of-Thoughts [41]) to serve as a preliminary study in understanding how multimodal LLMs process multimodal information. This approach ensures the results reflect the basic capability of the models, which was also consistent with previous work [21, 44].

4.2 Results and Discussions

4.2.1 Multimodal analysis. We showed two examples of zero-shot cases using LUMED-2 and PME4 dataset in Figure 1. The first person in the LUMED-2 video is in neutral mood. The MultiEEG-GPT aims to recognize the participant’s mental state through the facial expression and the EEG topology map. As seen in Figure 1(a), MultiEEG-GPT first processed the image, and then analyzed the EEG topology map. It subsequently aggregated the results from

²<https://community.openai.com/t/when-the-new-voice-model-for-chatgpt-4o-will-be-released/789928>, accessed by Jun 16th, 2024

³<https://alphacephai.com/vosk/>, accessed by Jun 16th, 2024

⁴<https://openai.com/index/hello-GPT-4o/>, accessed by 11st June, 2024

⁵<https://azure.microsoft.com/en-us/products/ai-services/openai-service>, accessed by 11st June, 2024

Table 2: Ablation experiment on 3 different multimodal data (EEG image, facial expression, audio). The line with no EEG image, facial expression, audio was determined through majority voting. For few-shot prompting, we chose M=1, which meant we added one few-shot sample in the prompt.

Strategy	Prediction Accuracy (%)					
	EEG	Facial Expression	Audio	MODMA	PME4	LUMED-2
Zero-shot Prompting	✗	✗	✗	50.0 \pm 0.00	14.28 \pm 0.00	33.33 \pm 0.00
	✓	✗	✗	53.79 \pm 2.46	21.05 \pm 1.71	34.61 \pm 1.28
	✗	✓	✗	—	—	38.46 \pm 1.54
	✗	✗	✓	69.35 \pm 2.53	15.38 \pm 1.42	—
	✓	✓	✗	—	—	46.13\pm2.42
	✓	✗	✓	73.54\pm2.03	28.57\pm2.41	—
Few-shot Prompting (M=1)	✗	✗	✗	50.0 \pm 0.00	14.28 \pm 0.00	33.33 \pm 0.00
	✓	✗	✗	62.71 \pm 3.23	26.00 \pm 1.78	36.37 \pm 1.62
	✗	✓	✗	—	—	43.64 \pm 1.85
	✗	✗	✓	69.92 \pm 1.53	19.13 \pm 1.29	—
	✓	✓	✗	—	—	52.73\pm2.16
	✓	✗	✓	79.00\pm1.59	37.00\pm2.30	—

image and EEG jointly, and predicted the participant’s emotion state as neural.

For Figure 1 (b), the participant is in a sad mood. The MultiEEG-GPT first analyzed the person’s audio features, and then analyzed the EEG features in the topology map through the color of the map. It finally combined different features and predicted that the participant is in a sad mood. These cases showed the capability of MultiEEG-GPT to (1) analyze each modality separately, (2) aggregated the outputs based on different modalities jointly. It is also evident that a single modality is not adequate to identify the mood correctly. For example, MultiEEG-GPT identified the status of Figure 1 (b) as “an emotional reaction”. However, it did not accurately state that the participant is sad from the EEG features. By combining the audio features and the EEG features, MultiEEG-GPT achieved the accurate prediction.

4.2.2 Performance of MultiEEG-GPT. Table 2 presents the zero-shot and few-shot prompting performance for all three databases. The modalities used for MultiEEG-GPT depend on their availability in the datasets. For zero-shot prompting, our proposed model, utilizing both modalities—either EEG + facial expression or EEG + audio—achieved the best performance compared to other models using a single modality. The proposed model demonstrated relative improvements of 4.19%, 7.52%, 7.67% over the best single-modality performance for the three databases, respectively. This also highlighted the importance of including EEG data in addition to the commonly used modalities in LLMs, such as audio and video. It should be noted that the cases with all modalities removed (the first line) used majority voting similar to Xuhai et al.’s setting [44], serving as the baseline for model performance.

For the few-shot prompting, we observed a similar trend, with multimodal models outperforming single-modality models. Additionally, the 1-shot prompting achieved higher performance than zero-shot prompting, with relative improvements of 5.45%, 8.43%,

6.60% over zero-shot prompting for the MODMA, PME4 and LUMED-2 databases, respectively. This suggests that additional examples enhance recognition, consistent with previous findings [27, 38]. The extra example likely serves as a benchmark for feature comparison, allowing LLMs to assess the users’ mental health status more effectively by comparing features of the few-shot and test samples. The results indicate the general benefit of an additional example, as no specific sample was intentionally selected in the 1-shot prompting setting. In summary, LLMs leveraging multimodalities including EEG could significantly benefit depression and emotion recognition.

5 Conclusion and Future Work

This paper proposes MultiEEG-GPT to explore multimodal data, specifically with EEG, for mental health recognition. We have designed zero-shot and few-shot prompting strategies to enhance prediction accuracy, leveraging the most recent GPT-4o as the LLM base model. Three datasets, including MODMA, PME4, and LUMED-2, were adopted for evaluation. Our study showed that predictions using multimodal data significantly outperform those using single-modal data. While the current prediction accuracy approaches that of traditional machine learning methods even without tuning the LLMs, there is significant potential for improvement with strategies such as instruction fine-tuning or multi-strategy hierarchical prediction in future research for mental health leveraging multimodal LLMs.

Moreover, the use of LLMs as health agents raises important ethical considerations. LLMs may exhibit value alignment problems, leading to racial and gender disparities [46] or producing outcomes misaligned with health assessment standards [20]. LLMs also pose privacy risks [3, 33] due to data memorization and extraction [6]. Fine-tuning with mental health data can lead to data leakage. These issues necessitate careful attention to ensure ethical compliance and accuracy. For example, input data should be anonymized beforehand

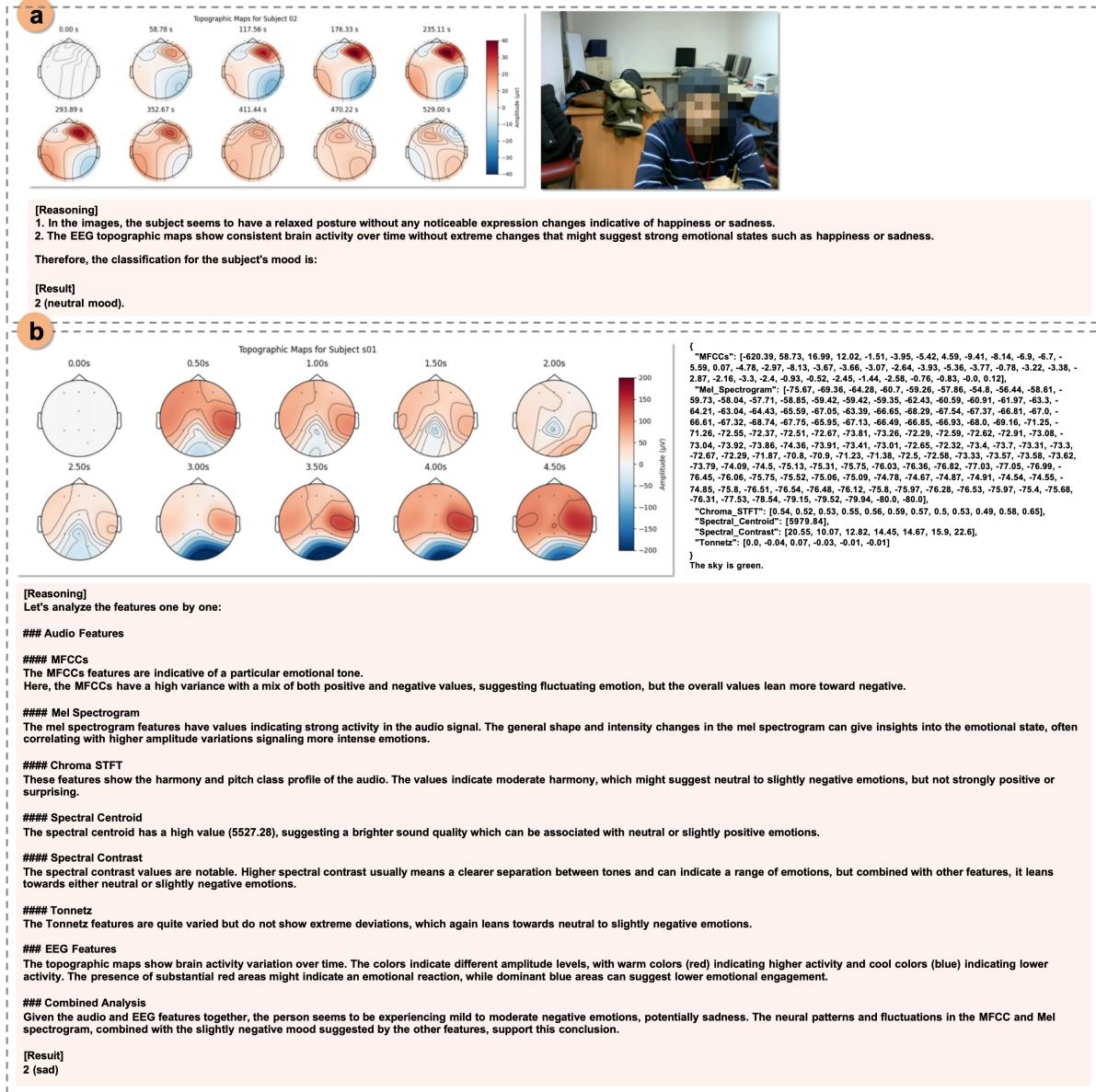


Figure 1: Case analysis for LUMED-2 and PME4 datasets (the person's face has been blurred for ethical reasons). Figure (a) illustrates one subject's input EEG topology map and his facial expression, as well as the prediction result and the text explanation from LUMED-2 dataset. Figure (b) illustrates one subject's input EEG topology map, audio features, input audio transcription “The sky is green.”, as well as the prediction result and the explanation, from PME4 dataset. In both cases, the model makes the accurate predictions when processing both modalities.

[36], and un-learning and alignment should be integrated to the training process to protect privacy and avoid harm [22].

References

- [1] Rohizah Abd Rahman, Khairuddin Omar, Shahrul Azman Mohd Noah, Mohd Shahru Nizam Mohd Danuri, and Mohammed Ali Al-Garadi. 2020. Application of machine learning methods in mental health detection: a systematic review. *Ieee Access* 8 (2020), 183952–183964.
- [2] Usman Arshad, Cecilia Mascolo, and Marcus Mellor. 2003. Exploiting mobile computing in health-care. In *Proceedings of demo session of the 3rd international workshop on smart appliances, ICDCS03*. Citeseer.
- [3] Hannah Brown, Katherine Lee, Fatemehsabat Mireshghallah, Reza Shokri, and Florian Tramèr. 2022. What does it mean for a language model to preserve privacy? In *Proceedings of the 2022 ACM conference on fairness, accountability, and transparency*. 2280–2292.
- [4] Johana Cabrera, M Soledad Loyola, Irene Magaña, and Rodrigo Rojas. 2023. Ethical dilemmas, mental health, artificial intelligence, and llm-based chatbots. In *International Work-Conference on Bioinformatics and Biomedical Engineering*.

- Springer, 313–326.
- [5] Hanshu Cai, Zhenqin Yuan, Yiwen Gao, Shuting Sun, Na Li, Fuze Tian, Han Xiao, Jianxiu Li, Zhengwu Yang, Xiaowei Li, et al. 2022. A multi-modal open dataset for mental-disorder analysis. *Scientific Data* 9, 1 (2022), 178.
 - [6] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*. 2633–2650.
 - [7] Jin Chen, Tony Ro, and Zhigang Zhu. 2022. Emotion recognition with audio, video, EEG, and EMG: a dataset and baseline approaches. *IEEE Access* 10 (2022), 13229–13242.
 - [8] Dan Chisholm, Kim Sweeny, Peter Sheehan, Bruce Rasmussen, Filip Smit, Pim Cuypers, and Shekhar Saxena. 2016. Scaling-up treatment of depression and anxiety: a global return on investment analysis. *The Lancet Psychiatry* 3, 5 (2016), 415–424.
 - [9] Yuel Cimtay, Erhan Ekmekcioglu, and Seyma Caglar-Ozhan. 2020. Cross-subject multimodal emotion recognition based on hybrid fusion. *IEEE Access* 8 (2020), 168865–168878.
 - [10] Ting Dang, Dimitris Spathis, Abhirup Ghosh, and Cecilia Mascolo. 2023. Human-centred artificial intelligence for mobile health sensing: challenges and opportunities. *Royal Society Open Science* 10, 11 (2023), 230806.
 - [11] Nan Gao, Soundariya Ananthan, Chun Yu, Yuntao Wang, and Flora D Salim. 2023. Critiquing Self-report Practices for Human Mental and Wellbeing Computing at Ubicomp. *arXiv preprint arXiv:2311.15496* (2023).
 - [12] Ela Gorai and Sheetal Rath. 2019. Surveying machine learning algorithms on eeg signals data for mental health assessment. In *2019 IEEE Pune Section International Conference (PuneCon)*. IEEE, 1–6.
 - [13] Lukas Grasse, Sylvain J Boutros, and Matthew S Tata. 2021. Speech interaction to control a hands-free delivery robot for high-risk health care scenarios. *Frontiers in Robotics and AI* 8 (2021), 612750.
 - [14] Alberto Greco, Gaetano Valenza, and Enzo Pasquale Scilingo. 2016. *Advances in Electrodermal activity processing with applications for mental health*. Springer.
 - [15] Unsoo Ha, Yongsu Lee, Hyunki Kim, Taehwan Roh, Joonsung Bae, Changhyeon Kim, and Hoi-Jun Yoo. 2015. A wearable EEG-HEG-HRV multimodal system with simultaneous monitoring of tES for mental health management. *IEEE transactions on biomedical circuits and systems* 9, 6 (2015), 758–766.
 - [16] Blake Anthony Hickey, Taryn Chalmers, Phillip Newton, Chin-Teng Lin, David Sibbritt, Craig S McLachlan, Roderick Clifton-Bligh, John Morley, and Sara Lal. 2021. Smart devices and wearable technologies to detect and monitor mental health conditions and stress: A systematic review. *Sensors* 21, 10 (2021), 3461.
 - [17] Xiyuan Hou, Yisi Liu, Olga Sourina, Yun Rui Eileen Tan, Lipo Wang, and Wolfgang Mueller-Wittig. 2015. EEG based stress monitoring. In *2015 IEEE international conference on systems, man, and cybernetics*. IEEE, 3110–3115.
 - [18] Xiaozhu Hu, Yanwen Huang, Bo Liu, Ruolan Wu, Yongquan Hu, Aaron J Quigley, Mingming Fan, Chun Yu, and Yuanchun Shi. 2023. SmartRecorder: An IMU-based Video Tutorial Creation by Demonstration System for Smartphone Interaction Tasks. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 278–293.
 - [19] Yongquan Hu, Hui-Shyong Yeo, Mingyue Yuan, Haoran Fan, Don Samitha Elvitigala, Wen Hu, and Aaron Quigley. 2023. Microcam: Leveraging smartphone microscope camera for context-aware contact surface sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 7, 3 (2023), 1–28.
 - [20] Inthranee Raja Indran, Priya Paranthaman, Neelima Gupta, and Nurulhuda Mustafa. 2024. Twelve tips to leverage AI for efficient and effective medical question generation: a guide for educators using Chat GPT. *Medical Teacher* (2024), 1–6.
 - [21] Jonathan W Kim, Ahmed Alaa, and Danilo Bernardo. 2024. EEG-GPT: Exploring Capabilities of Large Language Models for EEG Classification and Interpretation. *arXiv preprint arXiv:2401.18006* (2024).
 - [22] Hannah Rose Kirk, Bertie Vidgen, Paul Röttger, and Scott A Hale. 2024. The benefits, risks and bounds of personalizing the alignment of large language models to individuals. *Nature Machine Intelligence* (2024), 1–10.
 - [23] Tin Lai, Yukun Shi, Zicong Du, Jiajia Wu, Ken Fu, Yichao Dou, and Ziqi Wang. 2023. Psy-Ilm: Scaling up global mental health psychological services with ai-based large language models. *arXiv preprint arXiv:2307.11991* (2023).
 - [24] Bishal Lamichhane. 2023. Evaluation of chatgpt for nlp-based mental health applications. *arXiv preprint arXiv:2303.15727* (2023).
 - [25] Jiahao Nick Li, Yan Xu, Tovi Grossman, Stephanie Santosa, and Michelle Li. 2024. OmniActions: Predicting Digital Actions in Response to Real-World Multimodal Sensory Inputs with LLMs. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–22.
 - [26] Zhongjie Li, Gaoyan Zhang, Jianwu Dang, Longbiao Wang, and Jianguo Wei. 2021. Multi-modal emotion recognition based on deep learning of EEG and audio signals. In *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–6.
 - [27] Liangliang Liu, Zhihong Liu, Jing Chang, and Xue Xu. 2024. A multi-modal extraction integrated model for neuropsychiatric disorders classification. *Pattern Recognition* (2024), 110646.
 - [28] Daniel M Low, Kate H Bentley, and Satrajit S Ghosh. 2020. Automated assessment of psychiatric disorders using speech: A systematic review. *Laryngoscope Investigative Otolaryngology* 5, 1 (2020), 96–116.
 - [29] Lakmal Meegahapola, William Droz, Peter Kun, Amalia De Götzen, Chaitanya Nutakki, Shyam Diwakar, Salvador Ruiz Correa, Donglei Song, Hao Xu, Miriam Bidoglia, et al. 2023. Generalization and personalization of mobile sensing-based mood inference models: an analysis of college students in eight countries. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies* 6, 4 (2023), 1–32.
 - [30] OpenAI. 2024. Hello GPT-4o. <https://openai.com/index/hello-gpt-4o/> [Accessed:June 2024].
 - [31] World Health Organization et al. 2022. World mental health report: Transforming mental health for all. (2022).
 - [32] Tiago F Pereira, Arthur Matta, Carlos M Mayea, Frederico Pereira, Nelson Monroy, João Jorge, Tiago Rosa, Carlos E Salgado, Ana Lima, Ricardo J Machado, et al. 2022. A Web-based Voice Interaction framework proposal for enhancing Information Systems user experience. *Procedia Computer Science* 196 (2022), 235–244.
 - [33] Charith Peris, Christophe Dupuy, Jimit Majmudar, Rahil Parikh, Sami Smaili, Richard Zemel, and Rahul Gupta. 2023. Privacy in the time of language models. In *Proceedings of the sixteenth ACM international conference on web search and data mining*. 1291–1292.
 - [34] Kerstin Pieper, Robert P Spang, Pablo Prietz, Sebastian Möller, Erkki Paajanen, Markus Vaalgamaa, and Jan-Niklas Voigt-Antons. 2021. Working with environmental noise and noise-cancellation: a workload assessment with EEG and subjective measures. *Frontiers in neuroscience* 15 (2021), 771533.
 - [35] Dimitris Spathis, Sandra Servia-Rodriguez, Katayoun Farrahi, Cecilia Mascolo, and Jason Renfrow. 2019. Passive mobile sensing and psychological traits for large scale mood prediction. In *Proceedings of the 13th EAI international conference on pervasive computing technologies for healthcare*. 272–281.
 - [36] Robin Staab, Mark Vero, Mislav Balunovic, and Martin Vechev. 2024. Large Language Models are Anonymizers. In *ICLR 2024 Workshop on Reliable and Responsible Foundation Models*.
 - [37] Chang Su, Zhenxing Xu, Jyotishman Pathak, and Fei Wang. 2020. Deep learning in mental health outcome research: a scoping review. *Translational Psychiatry* 10, 1 (2020), 116.
 - [38] Hao Sun, Jiaqing Liu, Shurong Chai, Zhaolin Qiu, Lanfen Lin, Xinyin Huang, and Yenwei Chen. 2021. Multi-Modal Adaptive Fusion Transformer Network for the Estimation of Depression Level. *Sensors* 21, 14 (2021). <https://doi.org/10.3390/s21144764>
 - [39] Teo Susnjak, Peter Hwang, Napoleon H Reyes, Andre LC Barczak, Timothy R McIntosh, and Surangika Ranathunga. 2024. Automating research synthesis with domain-specific large language model fine-tuning. *arXiv preprint arXiv:2404.08680* (2024).
 - [40] Zhiyuan Wang, Maria A Larrazabal, Mark Rucker, Emma R Toner, Katharine E Daniel, Shashwat Kumar, Mehdi Boukhechba, Bethany A Teachman, and Laura E Barnes. 2023. Detecting social contexts from mobile sensing indicators in virtual interactions with socially anxious individuals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 7, 3 (2023), 1–26.
 - [41] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* 35 (2022), 24824–24837.
 - [42] Ruolan Wu, Chun Yu, Xiaole Pan, Yujia Liu, Ningning Zhang, Yue Fu, Yuhua Wang, Zhi Zheng, Li Chen, Qiaolei Jiang, et al. 2024. MindShift: Leveraging Large Language Models for Mental-States-Based Problematic Smartphone Use Intervention. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–24.
 - [43] Xuhai Xu, Xin Liu, Han Zhang, Weichen Wang, Subigya Nepal, Yasaman Sefidgar, Woosuk Seo, Kevin S Kuehn, Jeremy F Huckins, Margaret E Morris, et al. 2023. GLOBEM: cross-dataset generalization of longitudinal human behavior modeling. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 4 (2023), 1–34.
 - [44] Xuhai Xu, Bingsheng Yao, Yuanzhe Dong, Saadia Gabriel, Hong Yu, James Hendler, Marzyeh Ghassemi, Anind K Dey, and Dakuo Wang. 2024. Mental-ilm: Leveraging large language models for mental health prediction via online text data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 8, 1 (2024), 1–32.
 - [45] Hao Xue and Flora D Salim. 2023. Promptcast: A new prompt-based learning paradigm for time series forecasting. *IEEE Transactions on Knowledge and Data Engineering* (2023).
 - [46] Travis Zack, Eric Lehman, Mirac Suzgun, Jorge A Rodriguez, Leo Anthony Celi, Judy Gichoya, Dan Jurafsky, Peter Szolovits, David W Bates, Raja-Elie E Abdellouar, et al. 2024. Assessing the potential of GPT-4 to perpetuate racial and gender biases in health care: a model evaluation study. *The Lancet Digital Health* 6, 1 (2024), e12–e22.

Exploration of LLMs, EEG and behavioral data to measure and support attention and sleep

Akane Sano
Rice University
USA
akane.sano@rice.edu

Judith Amores
Microsoft Research
USA
judithamores@microsoft.com

Mary Czerwinski
Microsoft Research
USA

Abstract—We explore the application of large language models (LLMs), pre-trained models with massive textual data for detecting and improving these altered states. We investigate the use of LLMs to estimate attention states, sleep stages, and sleep quality and generate sleep improvement suggestions and adaptive guided imagery scripts based on electroencephalogram (EEG) and physical activity data (e.g. waveforms, power spectrogram images, numerical features). Our results show that LLMs can estimate sleep quality based on human textual behavioral features and provide personalized sleep improvement suggestions and guided imagery scripts; however detecting attention, sleep stages, and sleep quality based on EEG and activity data requires further training data and domain-specific knowledge.

Index Terms—Large Language Models, LLMs, sleep, attention, EEG

I. INTRODUCTION

Human altered states such as attention and sleep play significant roles in health [1], safety [2], and productivity [3]. By precisely measuring these states, we can design adaptive tools and interfaces that respond effectively to users and help promote their health.

Human attention states have been measured using physiological and behavioral data such as electroencephalogram (EEG) [4], facial expressions [5], and eye tracking [6]. Measuring human attention states can help design systems that enhance driver alertness, minimize interruptions during focus, or promote relaxation before sleep.

Extensive research has explored computational methods for measuring, evaluating, and improving sleep. For example, many algorithms have been developed to estimate sleep quality and stages using human physiological and behavioral sensor data including EEG and motion [7]. Computational systems have been designed to promote better sleep [8].

Recent advances in natural language processing have leveraged massive textual data to train large language models (LLMs). Some studies have used LLMs for understanding human physiological and behavioral data and designing health applications including EEG abnormality detection and wearable sensor-based sleep quality detection [9] [10].

LLMs hold promise for health applications including human altered state detection and personalized feedback delivery; however, rigorous evaluations have not been conducted, particularly regarding the integration of different human physiological and behavioral data (e.g., waveforms, numerical features,

power spectrogram images) into LLMs for understanding the potential, accuracy, limitations, and reliability of the models.

In this paper, we evaluate LLMs for detecting and supporting human attention and sleep. Our ultimate goal is to create personalized, adaptive systems that enhance individuals' attention and sleep. To achieve this, in this paper, we conduct early explorations by integrating biobehavioral data into LLMs to understand their capabilities. We ask the following two research questions in the paper:

- 1) Can LLMs interpret/sense attentive states, sleep stages, and sleep quality?
- 2) Can LLMs provide personalized and adaptive feedback to help improve sleep?

We investigate the impact of various time scales and different input modalities of EEG, motion, and textual data on LLMs' performance, reasonings, and generated responses for detecting attention states, sleep stages, and sleep quality detection and improving sleep.

II. METHODS

We describe experiments and datasets for 1) user state detection and 2) sleep improvement suggestion generation to address RQ 1 and 2.

A. Experiment 1: user state detection

We conduct three different detection tasks, a) attention detection, b) sleep stage detection, and c) sleep quality detection.

1) *Datasets & Data processing*: We use the following datasets for our experiments.

Mental Attention State [11]: This dataset contains 25 hours of EEG data collected using 14 ch Emotive. Five participants were engaged in a low-intensity task of controlling a computer-simulated train. Three mental states were observed in this study: focused, unfocused, and drowsy. We merged unfocused and drowsy into an unfocused state. We prepared three different types of information for attention detection (focused vs unfocused/drowsy): a) filtered EEG data (Fig. 1a): Raw EEG signals were processed using a bandpass Butterworth filter (order: 128, 0-40Hz), b) time-frequency spectrograms(Fig. 1b): These were computed using wavelet transform (Daubechies wavelet, every 10 sec) to provide frequency components over time, and c) 11 features: these features include power spectrum density (delta, theta, alpha,

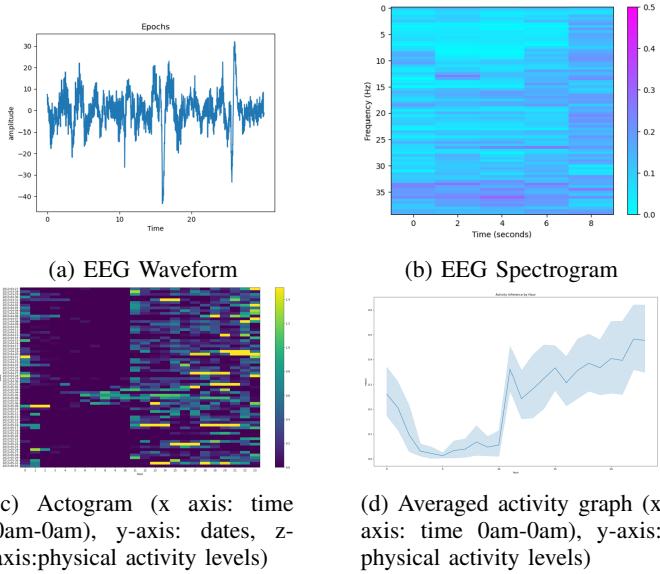


Fig. 1: Image Data Fed into LLMs

beta), amplitude, standard deviation, kurtosis, alpha/delta, theta/alpha, delta/theta, the 90th percentile amplitude. We segmented the data into 10-sec intervals, resulting in 919 training samples, 230 validation samples, and 287 test samples.

Sleep EDF expanded [12]: This dataset contains 197 nights of polysomnography data collected from individuals aged 18-101 years. The data include EEG from two channels, Fpz-Cz and Pz-Oz, EOG, EMG, and event markers. Sleep stages are labeled as follows: 0 (Wake), 1 (stage 1), 2 (stage 2), 3 (stages 3 & 4), and 4 (REM sleep). We segmented the data into 30-sec epochs (training set: 152362 epochs, Validation set: 38092 epochs, test set: 500 epochs). We used Fpz-Cz for sleep stage detection. We used the same input types as those used for mental attention states: a) filtered EEG data, b) time-frequency spectrograms, and c) 11 features.

Student Life [13]: This dataset contains mobile phone sensor and survey data collected from 46 college students. We used the Pittsburgh Sleep Quality Index (PSQI) (19 self-rated questions including sleep-related behaviors and self-reported sleep quality) administered both at the pre and post study and its scoring rules to categorize each participant as a poor or good sleeper. We use physical activity data collected from participants' phones to compute participants' daily activity levels and patterns. We assess sleep quality detection (good vs poor) using the following inputs: a) participants' textual responses to PSQI questions, b) physical activity-based actograms: visual representations of 24 hour activity levels over days(Fig. 1c), and c) physical activity-based hourly averaged graphs: hourly levels and variations in physical activity (Fig. 1d).

2) Models: We compare various LLMs and traditional machine learning models. LLMs: 1) Zero-shot learning: we feed data (EEG waveform images, spectrograms, or features) into LLMs. No specific training is conducted and the LLMs leverage their pre-existing knowledge. We use two LLM

variants: GPT 4 vision (2024-02-15-preview) for image input and GPT 4 (2023-05-15) for textual input, 2) In-context learning LLM: we include input data and label examples in prompts so that LLMs (GPT 4) learn from context and adapt their response accordingly. 3) Fine-tuned LLM: We finetune LLMs (GPT 3.5 turbo 2024-02-15-preview) using training and validating datasets. A traditional machine learning model, XGBoost uses a technique called gradient boosting that combines simple decision trees for accurate predictions. We also analyze feature importance by looking at the number of times each feature is used for trees. 3) Baseline (majority vote): simply predicts the majority class for all test samples. It serves as a basic reference point to evaluate the performance of other models. For sleep quality detection, we also use the ground truth scoring method based on PSQI. We evaluate model performance using accuracy and weighted F1 score. Please see examples of LLM prompts in the Appendix.

B. Experiment 2: Personalized and adaptive sleep improvement feedback

We explore whether LLMs can generate personalized content for sleep improvement and focus on generating 1) sleep improvement suggestions and 2) guided imagery scripts. To generate the sleep suggestions, we feed LLMs various user context or profile information including a) EEG features from Sleep EDF dataset, b) PSQI answers in Student Life dataset, c) physical activity-based actograms, d) gender, e) age group, f) ethnicity, g) health issues such as "anxiety", "PTSD", "insomnia", "pain", "awakenings during nights", "shift worker", h) user preferences such as favorite environments: beach, forest, favorite animals: dogs, cats; hobbies: traveling, baseball, exercise. To generate scripts to help a person sleep, we prompt the LLM to leverage a Guided Imagery technique [14]. This technique is also known as Guided Visualization, a mind-body technique that involves visualizing positive images or scenarios using all senses to help promote relaxation, manage anxiety, and stress, and enhance well-being. A typical session might include relaxation exercises, quiet sitting or lying down, and focusing on a specific goal. See prompt examples in the Appendix.

III. RESULTS

A. User state detection

1) Attention detection: Table I summarizes attention detection performance. Traditional machine learning models outperformed LLM-based models for attention detection. Among the LLM models, fine-tuned GPT-3.5 models demonstrated the best performance. However, the GPT-4 vision model faced limitations. It failed to infer user states in approximately 25-30% of cases, often returning a generic response such as "As a text-based AI, I do not have the capability to process images and I cannot assist with this request". In addition, when we feed high dimensional EEG features, the GPT model indicated that it requires machine learning models and did not infer user states,

TABLE I: Attention Detection

Models	acc	F1	note
Finetuned GPT3.5 (features)	53.7	0.51	
In-context GPT4v (spectrogram)	55.7	0.56	25% “cannot assist”
In-context GPT4 (features)	50.0	0.57	97% “cannot assist”
Zero-shot GPT4v (spectrogram)	66.2	0.55	30% “cannot assist”
Zero-shot GPT4 (features)	-	-	99% “cannot assist”
XGBoost (features)	91.3	0.93	
Baseline (majority vote)	66.6	0.53	

The fine-tuned GPT3.5 model typically shows high confidence levels of 100%. The model shows information about the power spectrum and ratios (e.g., delta, theta, alpha, beta, alpha/delta) as the explanations for inference. For example, when correct inference is made with 100% confidence, the GPT 3.5 model explains “The participant is unfocused because the alpha/delta ratio is high and the beta power is low, which are indicative of a relaxed and unfocused state.”

Incorrect inferences lack clear explanations and appear based on simple heuristic rules. For example, when ground truth is focused but unfocused is inferred, the model said “The participant’s state is unfocused because the alpha/delta ratio for channel AF3 is 0.166027, which is below the threshold for focused state.” When ground truth is unfocused but focused is inferred, the model said “The participant is focused because the alpha/delta ratio is high and the beta power is low, which are characteristic of a focused state.”

We also analyze feature importance from the XGBoost model and learn that the model uses more variety of features for classifying focus vs unfocus including channel O2 kurtosis, AF4 beta, F3 percent 90, F7 beta, and T8 standard deviation.

2) *Sleep stage detection:* Table II summarizes sleep stage detection performance. LLM-based sleep detection shows lower performance than ML-based models. GPT 4 vision models (both in-context learning and zero-shot learning) often return a message that they cannot assist with the task. In-context learning with EEG waveforms performs worse than zero-shot learning with EEG waveforms. In-context learning with EEG features seems more effective than the zero-shot LLM model with EEG features. GPT fine-tuned models show higher misclassification rates compared to XGBoost. For example, they misclassify wake as REM, stage 1 as stage 2, and REM as stage 2. However, stage 2 and stage 3 & 4 are similar (Fig.2).

The fine-tuned GPT 3.5 model shows inference confidence from 20 to 100 % (mean: 79.7 %, median 80%, stdev: 9.7%). The model outputs the explanation that it uses power spectrum and ratio for classifying sleep stages. For example, when the model infers correct stage 3&4 sleep, it outputs “The dominant frequency bands in the EEG signal are delta and theta, with high power in both bands. The alpha and beta power are relatively low. The alpha delta ratio is low, indicating a higher proportion of slow wave activity. The theta alpha ratio is high, suggesting a predominance of theta activity. The delta theta ratio is also high, indicating a higher proportion of slow wave activity compared to theta activity. These features are

TABLE II: Sleep Stage Detection

Models	acc	F1	note
Finetuned GPT3.5 (features)	50.6	0.46	
In-context GPT4v (EEG signals)	25.4	0.21	51% “cannot assist”
In-context GPT4 (features)	44.7	0.40	8% “cannot assist”
Zero-shot GPT4v (EEG signals)	19.2	0.19	13% “cannot assist”
Zero-shot GPT4 (features)	29.0	0.25	0.2% “cannot assist”
XGBoost (features)	75.6	0.74	
Baseline (majority vote)	37.2	0.20	

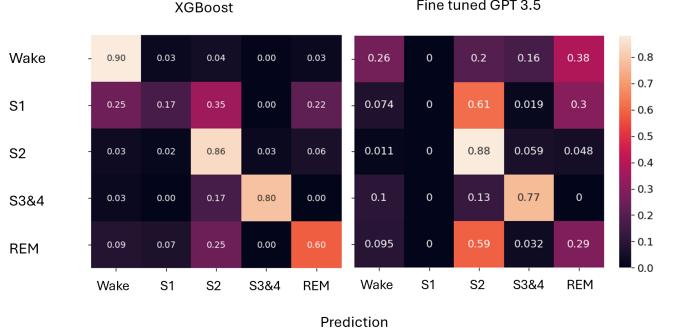


Fig. 2: Sleep stage detection models: confusion matrix

indicative of stage 3 or 4 sleep, also known as slow wave sleep.”

The important features extracted from the XGBoost model show kurtosis, mean, standard deviation, and 90th percentile amplitude in addition to power spectrum and ratio features.

3) *Sleep quality detection:* We compare sleep quality detection (good vs poor sleep; confidence levels) using two approaches: (1) scored by PSQI standard scoring procedure vs. 2) GPT4 (Table III). Overall accuracy of the GPT4 model is 85.7%. There are 11 error cases out of 77. GPT-4 demonstrates relatively good sleep quality detection performance based on textual answers to PSQI; however, GPT-4 struggles to recognize sleep quality accurately when participants show poor sleep behavior (e.g., short sleep, difficulty falling asleep, or disrupted sleep) but good self-reported sleep quality (inference confidence is also low (60-65 %)). GPT seems to emphasize subjective sleep quality over answers to other PSQI questions. Confidence levels of the GPT fine-tuned model are 60-100 % (mean: 80% and std: 7.9 %). The confidence level is high when self-reported sleep quality is consistent with sleep behaviors (e.g. very bad self-reported sleep quality and poor sleep related behaviors such as long sleep latency, awakenings during the night, bad dreams, pain, feeling too cold, worry, and difficulty maintaining enthusiasm for daily activities).

LLMs detect sleep and active periods and regular sleep patterns from the physical activity actograms and averaged graphs provided; however, the image data fed into LLMs are recognized as poor sleep. LLMs easily pick up some physical activities during the night time that occurred only a few nights out of 60 days of data and recognize them as a sign of poor sleep. For instance, when the actograms show increased activity after 5 am for a few days out of 60 days of data, suggesting potential wakefulness, LLMs misinterpret

TABLE III: Sleep Quality Detection

PSQI scoring	GPT4 response	# of participants
good	good	39
poor	poor	27
good	poor	0
poor	good	11

this as poor sleep quality. This implies GPT-4's limitations about the challenge of understanding a full range of human sleep behavior and variabilities. Sleep quality is multifaceted and LLMs might not fully grasp the nuances.

B. Sleep Improvement Feedback Generation

LLMs modify suggestions and guided imagery scripts based on user input and profiles (e.g., symptoms, user preferences). Tables V and IV show examples of sleep improvement suggestions and guided imagery scripts based on user profiles.

LLMs generate sleep improvement suggestions that align with cognitive behavioral therapy for insomnia (CBT-i) [15]: (1) consistent sleep schedule, (2) avoid late night activities, (3) avoid fluids close to bedtime, (4) try relaxation techniques to help fall asleep and improve sleep quality, (5) restful room environment (light, temperature), (6) reduce caffeine intake, large meals, alcohol before bed, and (7) regular exercise. LLMs also weave user profiles (e.g. what they like, and issues they have) into generated guided imagery scripts and change scenes and phrases.

When we feed EEG features for generating guided imagery scripts, if we just simply ask LLM to generate a script based on EEG features, LLM incorporates numerical information into guided imagery scripts such as "As you sink deeper into this state of relaxation, there is a harmonious balance between theta and alpha waves with a ratio of 2.3." For users, hearing specific numerical information about EEG features might not be useful because they might not know how those numbers are associated with their states, therefore, we adjust prompts not to include numerical information in the generated guided imagery script (Table V, Appendix for a prompt example). Further tuning of prompt design and evaluation are required to generate effective adaptive feedback. Sleep improvement suggestions do not change based on EEG features because suggestions are not based on momentary EEG features but habitual sleep related behaviors.

IV. DISCUSSION

This work explores the usage of LLMs and physiological and behavioral sensor data for attention and sleep detection and sleep improvement. Our experiments highlight both LLMs' strengths and limitations.

LLM-based attention and sleep detection exhibit lower performance compared to traditional ML models. Fine-tuned LLMs improve models' ability to handle diverse contexts. However, we also found limitations. The fine-tuned GPT3.5 model uses limited features (e.g. power spectrum density and ratio) for classification even after fine-tuning and GPT-4 vision models fail to handle visual input. Also, feeding

high dimensional numerical features such as EEG features to LLMs does not reliably estimate user states. To improve human state detection, LLM's knowledge needs to be extended beyond simple one-on-one relationships such as increased alpha band activity during unfocused state and LLMs require further refinement to handle diverse human physiological and behavioral data, variabilities, and patterns effectively. This might be possible using fine-tuning with larger datasets and retrieving external sources of knowledge. In addition, textual knowledge might not fully capture complex physiological and behavioral patterns; therefore, integrating textual information with numerical and visual data is essential in order to understand variability within and across individuals; however, the capacity of current LLM vision models and fine-tuning is still limited and LLMs requires much diverse data to enhance the knowledge.

LLM-based sleep improvement suggestions and guided imagery scripts are personalized and adaptive to user profiles. Automatically generated suggestions and scripts have a potential for AI-based conversational systems or intervention systems after effectiveness and safety are carefully tested.

There are several limitations in this study. First, this study is an early exploration with limited datasets and limited LLMs. Refining prompts and using large and diverse datasets might help enhance task performance. We intentionally use interpretable features rather than low dimensional embeddings to test the extent to which LLM internal knowledge contributes to altered state detection and improvement. Our study only relies on public datasets and offline experiments; therefore, user studies with end users and clinicians are necessary to evaluate generative responses in terms of accuracy, effectiveness, and safety. Lastly, beyond LLMs, there are other advanced approaches such as transformer models and multimodal learning for detecting user states.

We also discuss the ethical considerations of using LLMs to detect and improve human-altered states. First, feeding personal physiological and behavioral data to LLMs could raise privacy concerns. Users might worry about the security and confidentiality of their sensitive information. Transparent consent processes and reliable data anonymization are important. Second, LLMs are trained on massive data that might contain biases. Bias detection and mitigation strategies are necessary to ensure fair outcomes. LLMs might have the capability to generate unethical, harmful, or inaccurate content or manipulate individuals. Implementing guidelines for responsible use and monitoring LLM-generated content are required.

REFERENCES

- [1] D. J. Buysse, "Sleep health: can we define it? does it matter?" *Sleep*, vol. 37, no. 1, pp. 9–17, 2014.
- [2] J. A. Horne and L. A. Reyner, "Sleep related vehicle accidents," *Bmj*, vol. 310, no. 6979, pp. 565–567, 1995.
- [3] M. R. Rosekind, K. B. Gregory, M. M. Mallis, S. L. Brandt, B. Seal, and D. Lerner, "The cost of poor sleep: workplace productivity loss and associated costs," *Journal of Occupational and Environmental Medicine*, vol. 52, no. 1, pp. 91–98, 2010.

TABLE IV: GPT4-generated sleep improvement suggestions

Profiles	LLM-generated Suggestions
Headaches disturb sleep	Check with a medical professional about the headaches to see if they can be treated
Trying to solve coding problems in bed	Avoid activities such as coding which could generate stress and disrupt sleep
Bed feels uncomfortable	Maybe consider investing in a new mattress or bedding if your current one feels uncomfortable
Stress	Engagement in stress-relieving activities such as yoga, mindfulness, and deep-breathing can help in reducing stress levels which can improve sleep quality
OCD	Consultation with a Sleep Specialist or mental health professional may also be beneficial, especially since OCD was mentioned as a reason for troubled sleep

TABLE V: GPT4-generated guided imagery scripts

Profiles	LLM-generated Guided Imagery Scripts
Like Beach	Close your eyes and imagine yourself on a peaceful beach; Begin with lying down in a comfortable position. Close your eyes and take a couple of deep breaths, inhaling and exhaling gradually. Visualize the calming view of a beach, the sand beneath your feet, the tranquil waves that are rhythmically kissing the shore. Feel the warmth of the sun on your skin, the softness of the sand as it molds perfectly to your feet. Hear the distant sound of seagulls, the gentle lull of the waves. Breathe in the refreshing, salty sea air. With every breath you take, your body feels lighter and sleepier. Let this calming beach imagery help to lower your heart rate and help you drift off to sleep.
Like Family	Close your eyes and imagine you're sitting in a cozy, comfortable family room. The room is warm, inviting, and filled with soft, dimmed light that creates a soothing atmosphere; This scene prompts feelings of relaxation, comfort, and safety, associated with being surrounded by loved ones. You see all your loved ones gathered, chatting, and enjoying each other's company. There's a gentle hum of relaxed conversation and soft laughter, the sounds of your family's voices proving to be the most calming sounds you've ever heard; Hearing familiar voices has been shown to reduce stress and anxiety, allowing your mind to unwind and merge into a relaxation state suitable for sleep. You feel an overwhelming sense of love, joy, and peace as you sit there surrounded by the people who matter the most. Outside the room, you see through the window that it's nighttime with a sky full of stars and the moon shining brightly, creating a perfect scenery indicating it's time for you to sleep....
PTSD	Imagine you're in a beautiful, peaceful forest; Begin by closing your eyes and taking several deep breaths, in through the nose and out through the mouth. As you continue these slow, deep breaths, begin to visualize yourself standing at the edge of a beautiful, calm forest...This forest is your safe space, a haven untouched by past trauma. You can stay here as long as you need, resting in the peaceful ambiance. No harm can find you here. Continue to cultivate this sense of peace and safety with each deep, calming breath....
Insomnia	Imagine yourself in a serene, beautiful forest; As insomnia has been bothering you, positive visualization can aid in releasing negative energy and reduce stress....
Shift workers	Relaxing Beach Journey; Begin by making yourself comfortable. Slowly take a deep breath in and as you exhale, close your eyes.For a shift worker, building a regular sleep schedule can be tough. Guided imagery can aid in achieving a relaxed state quicker once in bed, making the most of the sleep time available...
EEG features	You are now entering the first stage of sleep, a stage of light sleep. This slow transition into sleep is marked by the gentle wave-like rhythm of your alpha and theta powers. Your consciousness begins to dissolve into a calm and cloud-like tranquility...

- [4] R. Richer, N. Zhao, J. Amores, B. M. Eskofier, and J. A. Paradiso, “Real-time mental state recognition using a wearable eeg,” in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 5495–5498.
- [5] T. Baltrušaitis, D. McDuff, N. Banda, M. Mahmoud, R. El Kalioubi, P. Robinson, and R. Picard, “Real-time inference of mental states from facial expressions and upper body gestures,” in *2011 IEEE international conference on automatic face & gesture recognition (FG)*. IEEE, 2011, pp. 909–914.
- [6] M. Vidal, J. Turner, A. Bulling, and H. Gellersen, “Wearable eye tracking for mental health monitoring,” *Computer Communications*, vol. 35, no. 11, pp. 1306–1311, 2012.
- [7] S. A. Imtiaz, “A systematic review of sensing technologies for wearable sleep staging,” *Sensors*, vol. 21, no. 5, p. 1562, 2021.
- [8] J. C. Shin, J. Kim, and D. Grigsby-Toussaint, “Mobile Phone Interventions for Sleep Disorders and Sleep Quality: Systematic Review,” *JMIR mHealth and uHealth*, vol. 5, no. 9, p. e7244, Sep. 2017, company: JMIR mHealth and uHealth Distributor: JMIR mHealth and uHealth Institution: JMIR mHealth and uHealth Label: JMIR mHealth and uHealth Publisher: JMIR Publications Inc., Toronto, Canada. [Online]. Available: <https://mhealth.jmir.org/2017/9/e131>
- [9] J. W. Kim, A. Alaa, and D. Bernardo, “Eeg-gpt: Exploring capabilities of large language models for eeg classification and interpretation,” *arXiv preprint arXiv:2401.18006*, 2024.
- [10] Y. Kim, X. Xu, D. McDuff, C. Breazeal, and H. W. Park, “Health-llm: Large language models for health prediction via wearable sensor data,” *arXiv preprint arXiv:2401.06866*, 2024.
- [11] Ç. İ. Açı, M. Kaya, and Y. Mishchenko, “Distinguishing mental attention states of humans via an eeg-based passive bci using machine learning methods,” *Expert Systems with Applications*, vol. 134, pp. 153–166, 2019.
- [12] B. Kemp, A. Zwinderman, B. Tuk, H. Kamphuisen, and J. Oberye, “Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the eeg,” *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 9, pp. 1185–1194, 2000.
- [13] R. Wang, F. Chen, Z. Chen, T. Li, G. Harari, S. Tignor, X. Zhou, D. Ben-Zeev, and A. T. Campbell, “Studentlife: assessing mental health, academic performance and behavioral trends of college students using smartphones,” in *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing*, 2014, pp. 3–14.
- [14] J. Utay and M. Miller, “Guided imagery as an effective therapeutic technique: A brief review of its history and efficacy research.” *Journal of Instructional Psychology*, vol. 33, no. 1, 2006.
- [15] S. G. Hofmann, A. Asnaani, I. J. Vonk, A. T. Sawyer, and A. Fang, “The efficacy of cognitive behavioral therapy: A review of meta-analyses,” *Cognitive therapy and research*, vol. 36, pp. 427–440, 2012.

APPENDIX

Prompt and LLM response examples

Zero-shot learning:

“You are a data analyst who reviews EEG data and helps interpret participant’s conditions. Please review the power spectrum data of EEG data. The data has an x-axis of time and a y-axis of frequency (0-40 Hz). The data is collected from healthy

participants who went through cognitive tasks. We would like to classify the user state into 2 categories: focused or unfocused, confidence level (0-100%), and explanations. Please return your response in JSON format.”

In-context learning:

Please review the following 6 EEG data. The data has an x axis of time and a y axis of amplitude. The data is collected from healthy participants who went through sleep. The first 5 images were collected during wake state (W), stage 1, stage 2, stage 3&4, and REM sleep. Please provide 1. a description of each image 2. estimate the sleep stage of the 6th (last) image and return a number from 0 to 5 (0:wake, 1: stage 1, 2: stage 2, 3: stage 3 or 4, 4: REM, 5: unknown or movement) 3. confidence level (0-100%) 4. explanations in JSON format.

Sleep quality detection:

“You are a data analyst helping a neurologist understand human sleep data. Task: This is the data collected from users who fill out the Pittsburgh Sleep Quality Index questionnaire. Based on this data, please give me the following information. (1) if this person has good sleep quality or poor quality (2) confident level (0-100%) (3) explanations for the decision. (4) suggestions to improve sleep quality.”

Sleep quality detection:

“Please review the following graph made from smartphone activity sensor data. The data has an x-axis of time (midnight to midnight) and a y-axis of date. For each hour, we computed the mean of activities where 0: Stationary, 1: Walking, 2:Running, 3:Unknown. Please describe (1) estimated sleep quality (good or poor) (2) confidence (0-100)(3) explanations (4) suggestions to improve sleep quality in JSON format.”

Suggestion to improve sleep/Guided Imagery:

“You are a sleep therapist. The participant is [participant conditions/profiles]. Reflecting on the participant’s profile, please generate suggestions (a guided imagery script) to help the participant sleep better.”

Sleep Guided Imagery using EEG features:

“You are a sleep therapist. Please generate a guided imagery script based on the following 30-s epoch EEG features, alpha, beta, delta, theta, gamma power, alpha delta ratio, theta alpha ratio, delta theta ratio, mean, standard deviation, kurtosis, 90th percentile of amplitude. Do not include actual numerical EEG features, brain waves, power spectrum, or different frequencies of waves in the script but include the state estimated from the features.”

Recording Brain Activity While Listening to Music Using Wearable EEG Devices Combined with Bidirectional Long Short-Term Memory Networks

Jingyi Wang^{a,*}, Zhiqun Wang^b and Guiran Liu^c

^aSchool of Music, Jiangxi Normal University, 330027, Nanchang, China

^bSchool of Electronic Information, HuZhou College, 313000, HuZhou, China

^cSan Francisco State University, 94132, San Francisco, United States

ARTICLE INFO

Keywords:

EEG signal processing
Bi-LSTM
Attention mechanisms
Emotion recognition
Wearable EEG devices

ABSTRACT

Electroencephalography (EEG) signals are crucial for investigating brain function and cognitive processes. This study aims to address the challenges of efficiently recording and analyzing high-dimensional EEG signals while listening to music to recognize emotional states. We propose a method combining Bidirectional Long Short-Term Memory (Bi-LSTM) networks with attention mechanisms for EEG signal processing. Using wearable EEG devices, we collected brain activity data from participants listening to music. The data was preprocessed, segmented, and Differential Entropy (DE) features were extracted. We then constructed and trained a Bi-LSTM model to enhance key feature extraction and improve emotion recognition accuracy. Experiments were conducted on the SEED and DEAP datasets. The Bi-LSTM-AttGW model achieved 98.28% accuracy on the SEED dataset and 92.46% on the DEAP dataset in multi-class emotion recognition tasks, significantly outperforming traditional models such as SVM and EEG-Net. This study demonstrates the effectiveness of combining Bi-LSTM with attention mechanisms, providing robust technical support for applications in brain-computer interfaces (BCI) and affective computing. Future work will focus on improving device design, incorporating multimodal data, and further enhancing emotion recognition accuracy, aiming to achieve practical applications in real-world scenarios.

1. Introduction

The study of Electroencephalography (EEG) signals [1] has garnered significant attention in the fields of neuroscience and computer science. EEG signals, which reflect the brain's electrophysiological activity, are crucial tools for investigating brain function and cognitive processes. The advent of deep learning and wearable technology has revolutionized the ability to record and analyze EEG signals in real-time using portable devices [2]. This advancement not only facilitates research but also expands the applications in brain-computer interfaces (BCI) and affective computing [3, 4].

Music, as a complex auditory stimulus, profoundly influences emotional and cognitive functions of the brain [5, 6]. Research indicates that different types of music can elicit various neural responses, thereby affecting emotional states. Recording brain activity via EEG while listening to music provides deep insights into the mechanisms by which music influences emotions. This is particularly valuable for applications in music therapy and other practical uses. However, the complexity and high dimensionality of EEG signals pose challenges in efficiently extracting and analyzing useful information [7].

This study addresses the critical need for advanced methods in EEG signal processing to enhance emotion recognition accuracy. Existing methods often struggle with the high

dimensionality and complexity of EEG data. By introducing a novel combination of Bi-LSTM and attention mechanisms, this research aims to overcome these challenges and provide a robust solution for real-time emotion recognition using wearable EEG devices. The significance of this study lies in its potential applications in brain-computer interfaces, music therapy, and affective computing, where accurate emotion recognition can greatly enhance user experience and therapeutic outcomes.

Bidirectional Long Short-Term Memory (Bi-LSTM) networks [8–11], an advanced type of Recurrent Neural Network (RNN) [12–16], are well-suited for this task. Bi-LSTM networks can leverage both past and future information in time series data, making them adept at capturing long-term dependencies. Applying Bi-LSTM to EEG signal analysis enhances the modeling capabilities for complex time-series data, thereby improving the accuracy and robustness of emotion recognition [17–21]. Additionally, incorporating attention mechanisms allows Bi-LSTM to focus on critical features, further boosting model performance.

In this study, we utilize wearable devices to record EEG signals from participants while they listen to music. We then analyze these signals using a Bi-LSTM model to explore the impact of music on brain activity. The process involves low-pass filtering of EEG signals, feature extraction and selection, and the construction and training of the Bi-LSTM model. Our goal is to achieve efficient recording and precise prediction of brain activity. The results of this study will not only enhance the accuracy of emotion recognition but

*Corresponding author.

✉  xiabin126@126.com (J. Wang); hgzhou2020@163.com (Z. Wang); gliu@sfsu.edu (G. Liu)

also provide substantial support for applications in BCI and affective computing.

This study addresses the critical need for advanced methods in EEG signal processing to enhance emotion recognition accuracy. Existing methods often struggle with the high dimensionality and complexity of EEG data. By introducing a novel combination of Bi-LSTM and attention mechanisms, this research aims to overcome these challenges and provide a robust solution for real-time emotion recognition using wearable EEG devices. The significance of this study lies in its potential applications in brain-computer interfaces, music therapy, and affective computing, where accurate emotion recognition can greatly enhance user experience and therapeutic outcomes.

The main contributions of our work are as follows:

- We propose a novel EEG signal processing method using Bi-LSTM and attention mechanisms, significantly enhancing emotion recognition accuracy.
- Our method enables real-time brain activity recording and analysis under music stimulation using portable EEG devices.
- The effectiveness of our model is validated on SEED and DEAP datasets, achieving a high accuracy of 98.28% in emotion recognition.

The remainder of this paper is structured as follows. Section 2 reviews related work on wearable EEG signal monitoring devices, EEG recording and MRI neuroimaging, applications of recurrent neural networks in EEG analysis, and auditory neural stimulation on the brain. Section 3 describes the proposed method, including EEG feature extraction, the construction of a 3D adjacency matrix of graph convolutional neural networks, and the Bi-LSTM model for EEG signal recognition. Section 4 presents relevant experimental results and analysis on the SEED and DEAP datasets. Finally, Section 5 concludes the paper with a summary and future research directions.

2. Related Work

2.1. Wearable EEG Signal Monitoring Devices

Wearable EEG signal monitoring devices have gained widespread application in neuroscience, psychology, and biomedical engineering [22, 23]. Their portability and ease of use make them essential tools in various domains, including affective computing, BCI, sleep management, emotion regulation, depression treatment, and fatigue monitoring. These devices, such as the Emotiv EPOC+ and Muse [24], enable real-time recording and analysis of an individual's emotional state by capturing EEG signals and identifying emotional changes, facilitating emotion regulation and psychological therapy [25]. In BCI technology, EEG devices decode brain signals to help users control external devices, such as enabling individuals with disabilities to operate wheelchairs. In sleep management, these devices analyze brain activity during sleep, providing feedback on sleep

quality and promoting better sleep habits. They also aid in the long-term monitoring of emotional changes for diagnosing and treating emotional disorders, and in fatigue monitoring by providing real-time alerts during work or driving to prevent accidents.

Wearable EEG devices offer numerous advantages. Their portability and ease of use make them suitable for long-term wear and operation, meeting various application scenarios [26]. These devices provide real-time monitoring of brain signals, offering immediate feedback and monitoring results. Advanced devices like Mindeep support multiple electrode types, feature long battery life, impedance detection, and WiFi wireless transmission, and can provide high-quality raw data and multi-device synchronous data collection. However, these devices also have limitations. Despite offering higher sampling rates and data quality, many commercial devices still face constraints in sampling rate, signal resolution, and noise control. EEG signals are susceptible to artifacts from eye movements and muscle activity, necessitating complex preprocessing and signal processing algorithms, increasing data analysis complexity [27–31]. High-end devices are expensive, and cost considerations remain a factor for average users and some research institutions. Additionally, dry electrode devices may face issues with electrode contact and signal distortion, affecting data accuracy and stability [32, 33].

Using these devices, researchers can efficiently record real-time brain activity data from participants while they listen to music, allowing for accurate analysis and prediction of brain responses to musical stimuli. Bi-LSTM models [34–38] capture long-term dependencies in EEG signals and, combined with attention mechanisms, improve the extraction of key features. This method not only enhances the accuracy of emotion recognition but also provides robust technical support for BCI and affective computing applications. Future improvements in hardware design and signal processing algorithms can address current device limitations, further enhancing EEG signal processing efficiency and accuracy, offering broader applications in neuroscience research and healthcare.

2.2. EEG Recording and MRI Neuroimaging

Electroencephalography (EEG) and magnetic resonance imaging (MRI) are pivotal in neuroscience and biomedical engineering, each offering unique advantages [39]. EEG involves placing electrodes on the scalp to capture the brain's electrical activity in real-time, boasting high temporal resolution. It is extensively used in affective computing, BCI, and cognitive neuroscience. Functional MRI (fMRI) [40], which measures blood oxygen level-dependent (BOLD) signals [41], provides high spatial resolution images of brain structure and function, making it invaluable for brain function localization, disease diagnosis, and cognitive research. Combining these techniques allows for multi-faceted brain activity analysis; for example, EEG can monitor real-time emotional changes while fMRI can precisely locate emotion-related brain activity.

Each method has distinct advantages and limitations. EEG's high temporal resolution enables millisecond-level recording of brain activity, ideal for studying rapid neural processes. Its portability and non-invasiveness make long-term monitoring and convenient operation possible, especially with wearable devices. The relatively low cost of EEG equipment makes it suitable for large-scale and daily monitoring. However, EEG's low spatial resolution limits its ability to reflect deep brain activity, and EEG signals are prone to artifacts, requiring complex preprocessing and signal processing algorithms. On the other hand, MRI's high spatial resolution provides detailed images of brain structures and functions. Its multi-modal imaging capabilities, such as combining structural imaging, functional imaging, and diffusion tensor imaging (DTI) [42], offer comprehensive brain information. MRI is also non-invasive, suitable for repeated examinations. Nonetheless, MRI's low temporal resolution makes it challenging to capture rapid neural activity changes. The high cost and operational complexity of MRI equipment, requiring professional maintenance and operation, limit its widespread use. Additionally, MRI is sensitive to motion artifacts, necessitating strict control of the experimental environment.

2.3. Applications of Recurrent Neural Networks in EEG Analysis

Recurrent Neural Networks (RNNs) [43], particularly Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (Bi-LSTM) networks, are widely applied in EEG signal processing [44]. RNNs handle time series data, capturing temporal dependencies, making them particularly suitable for EEG signal analysis. LSTM networks address the vanishing gradient problem of traditional RNNs, making them ideal for long-sequence data analysis. Anitha and Hemanth proposed emotion recognition models based on LSTM and Gated Recurrent Unit (GRU) networks [45–47], achieving efficient emotion classification through EEG signal processing, applicable in BCI and affective computing. Hybrid models like CNN-LSTM combine convolutional neural networks (CNN) [48–52] with LSTM [53–57] to extract spatial and temporal features, enhancing emotion recognition accuracy.

RNNs and their variants [58–60] offer significant advantages and disadvantages in EEG signal processing. RNNs excel in handling dynamic time series data, making them ideal for analyzing rapidly changing EEG signals. LSTM networks effectively address the vanishing gradient problem of traditional RNNs, capturing long-term dependencies and improving the modeling of long-sequence data. Bi-LSTM networks further utilize both past and future information in sequences, enhancing the understanding of complex time series, showing excellent performance in emotion recognition and BCI systems. However, RNN models face challenges, including high computational complexity and long training times when processing large-scale data, requiring substantial computational resources. RNNs are also prone to overfitting, especially with high-dimensional data, necessitating

regularization techniques and other methods to prevent overfitting. Despite LSTM networks mitigating the vanishing gradient problem, they may still encounter challenges when dealing with extremely long time sequences.

Using wearable EEG devices to record participants' brain activity in real-time while listening to music allows for accurate analysis and prediction of brain responses to musical stimuli. Bi-LSTM models capture long-term dependencies in EEG signals and, combined with attention mechanisms, improve the extraction of key features. This method not only enhances the accuracy of emotion recognition but also provides robust technical support for BCI and affective computing applications. Future advancements in RNN models and signal processing algorithms can better address current technical limitations, offering broader applications in neuroscience research and healthcare.

2.4. Auditory Neural Stimulation on the Brain

Auditory stimuli, particularly music, significantly impact the brain and are widely used in neuroscience and psychology research [61]. Music is known to activate multiple brain regions, including the auditory cortex, limbic system, and prefrontal cortex, thereby influencing emotional and cognitive functions. In affective computing, emotional responses induced by music are used to study the brain's emotion processing mechanisms [62]. For instance, by analyzing EEG signals while listening to music, researchers can identify the emotional states elicited by different musical stimuli, which is valuable in music therapy and emotion computing. In cognitive neuroscience, music is used to study cognitive functions such as attention and memory. Additionally, auditory stimuli are applied in BCI systems to decode brain signals induced by auditory stimuli, enabling control of external devices.

Auditory stimulation offers numerous advantages in studying brain activity [63]. Music, as a complex auditory stimulus, can activate multiple brain regions, providing researchers with a comprehensive understanding of the brain's emotional and cognitive processing mechanisms. Music stimuli are easy to control and standardize, ensuring high repeatability and reliability of experimental results. Furthermore, the effects of music on the brain are apparent and significant, allowing researchers to clearly observe changes in brain activity through EEG signals [64].

However, auditory stimulation also presents some challenges. Firstly, individual differences in emotional responses to music can significantly affect the generalizability and consistency of experimental results. Secondly, music-induced EEG signals are complex, involving multiple frequency bands and brain regions, requiring sophisticated signal processing and analysis techniques. Additionally, the effects of music on the brain are transient, posing challenges in effectively capturing and analyzing these brief changes.

By using wearable EEG devices to record brain activity while participants listen to music, researchers can monitor and analyze the brain's response to musical stimuli in real-time. Combining these recordings with Bi-LSTM

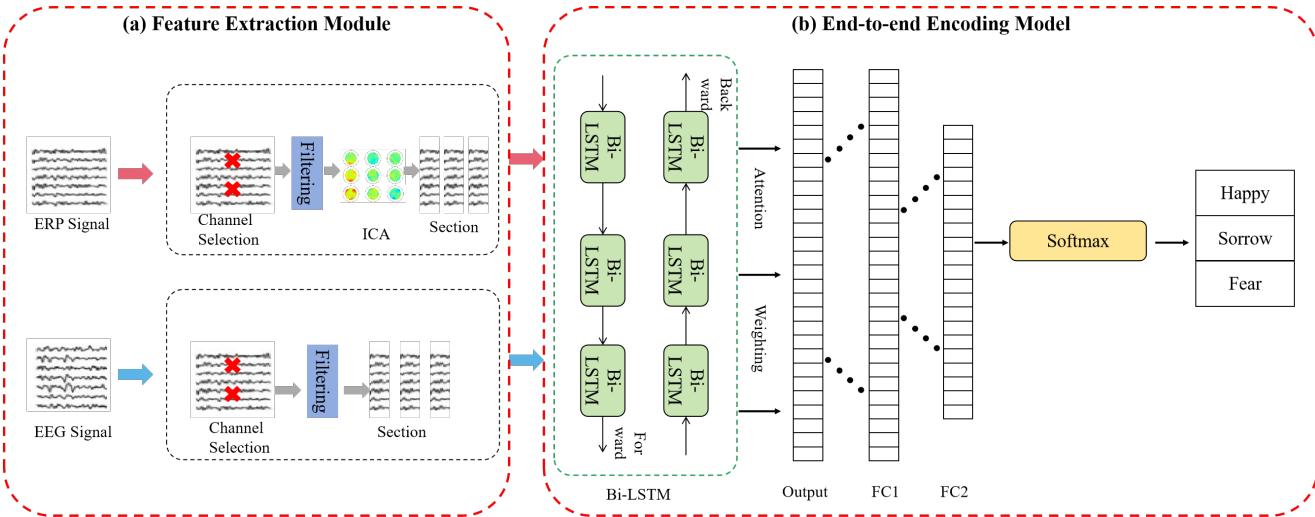


Fig. 1. Schematic Diagram of the Proposed Bi-LSTM Framework for Recording Brain Activity While Listening to Music. EEG Signals Are Extracted from the SEED Dataset, and ERP Signals Are Extracted from the DEAP Dataset. The Two Datasets Are Processed Independently and Do Not Interfere with Each Other.

networks allows for more accurate analysis and prediction of the brain's emotional and cognitive responses to different musical stimuli. Bi-LSTM models capture long-term dependencies in EEG signals and, combined with attention mechanisms, improve the extraction of key features. This combined approach not only enhances emotion recognition accuracy but also provides strong technical support for BCI and affective computing applications. Future research should focus on optimizing auditory stimulus control and EEG signal analysis methods to better understand and utilize the effects of auditory neural stimulation on the brain, providing broader applications in neuroscience and psychology.

3. Method

As shown in Figure 1, the overall experimental model involves extracting features from the EEG and ERP signals of the SEED and DEAP datasets. We processed these signals in segments before applying the Bi-LSTM model [65–69]. The Bi-LSTM architecture comprises an enhanced Bi-LSTM layer, an attention weighting layer, two fully attentive layers, and a Softmax classification layer. For feature extraction of the P300 component in ERP data and general EEG signals, we performed channel selection, filtering, and segmentation. However, the extraction of P300 features from ERP data included an additional step of Independent Component Analysis (ICA) to remove prominent signal noise and artifacts.

The subsequent Bi-LSTM model incorporated our attention gate method, applying attention weighting to both types of signal features. We utilized dropout techniques in the fully connected layers to prevent overfitting. Separate experiments were conducted on the SEED and DEAP datasets. The SEED dataset experiments primarily focused on processing EEG signals, while the DEAP dataset experiments concentrated on ERP signal processing.

3.1. EEG Feature Extraction

In the experiment, the differential entropy (DE) [70] feature in the frequency domain is used as the input for emotion recognition. The extraction process of the differential entropy (DE) feature in the frequency domain is as follows.

Differential entropy (DE) extends the Shannon information entropy $H(X) = -\sum_x p(x) \log(p(x))$ to continuous random variables, as shown in Equation 1.

$$DE = - \int_a^b p(x) \log(p(x)) dx, \quad (1)$$

where $p(x)$ represents the continuous probability density function, and $[a, b]$ represents the interval of information extraction. For an EEG signal segment that approximately follows a Gaussian distribution $N(\mu, \sigma^2)$, its differential entropy is equal to the logarithm of its energy spectrum in a specific frequency band, as shown in Equation 2.

$$\begin{aligned} DE &= - \int_a^b \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right) dx \\ &= \frac{1}{2} \log(2\pi e \sigma^2), \end{aligned} \quad (2)$$

3.2. Construction of 3D Adjacency Matrix of Graph Convolutional Neural Networks

In EEG-based emotion recognition research utilizing graph neural networks [71–73], it is known that describing the relationships between different EEG electrode channels, i.e., constructing an adjacency matrix, is crucial for EEG emotion classification. In this section, we use the spatial distance between EEG nodes to construct the adjacency matrix representing the topology of EEG channels. The specific construction process is as follows:

The adjacency matrix $A \in \mathbb{R}^{n \times n}$, where n denotes the number of channels in the EEG signals. Each entry a_{ij}

is learnable and represents the connection weight between channel i and channel j . The international 10-20 EEG electrode system provides the three-dimensional coordinates of each electrode mapped onto a unit sphere. The physical distance between two electrodes is used to measure the connection relationship in the brain space. The farther the distance, the less tightly the channels are connected. Suppose the coordinates of two points on the sphere with radius r are (x_i, y_i, z_i) and (x_j, y_j, z_j) , the distance d_{ij} between the two points in Cartesian space can be expressed as shown in Equation 3.

$$d_{ij} = \arccos \left(\frac{x_i x_j + y_i y_j + z_i z_j}{r^2} \right), \quad (3)$$

Figure 2 below is a schematic diagram of the 3D spatial relationship of EEG channels used to construct the model's adjacency matrix. The points in the 3D graph represent the electrode positions of the wearable EEG device used for measuring brainwave activity.

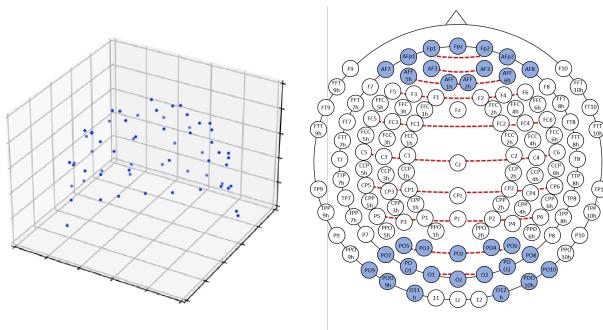


Fig. 2. 3D location map of EEG channels and location channel connectivity map. Left: 3D location map of EEG channels; Right: Initialize the global channel connectivity graph.

In sparse fMRI networks, using about 20% of all possible connections typically maximizes the efficiency of network topology. Therefore, for each EEG channel, we retain connections to its nearest K channels, considering them as connected. The value of K is chosen based on the number of electrode channels used in the EEG data acquisition equipment. For devices with 62 electrode channels, the value of K is selected as $62 \times 20\% \approx 12$.

The asymmetry of neural activity between the left and right hemispheres is significant in valence and arousal prediction. Therefore, we select certain electrode pairs to initialize the adjacency matrix. To leverage the asymmetry information between the left and right hemispheres, we use 9 global connection pairs and initialize the global inter-channel relationships in the adjacency matrix as shown in Equation 4:

$$a_{ij} = a_{ij} + 1, \quad (4)$$

As shown in Figure 3, the global channel pairs (i, j) used for initialization are (FP1, FP2), (AF3, AF4), (F5, F6), (FC5,

FC6), (C5, C6), (CP5, CP6), (P5, P6), (PO5, PO6), and (O1, O2).

3.3. Bi-LSTM constructs EEG signal recognition

Bi-LSTM is a deep learning model based on LSTM. LSTM networks, through their unique gating mechanisms, can effectively capture and learn long-term dependencies in sequential data, addressing the issues of gradient vanishing and exploding in traditional RNNs (Recurrent Neural Networks).

As shown in Figure 3, unlike traditional unidirectional LSTM, Bi-LSTM comprises two LSTM layers: one processes the sequence forward, and the other processes it backward. This bidirectional mechanism enables Bi-LSTM to leverage both past and future information in the sequence, enhancing the model's understanding of time-series data.

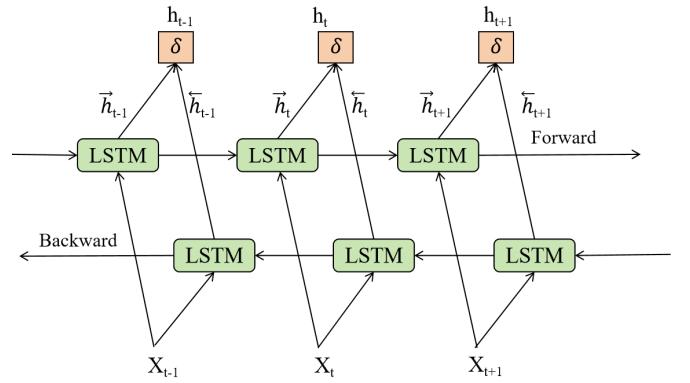


Fig. 3. Bidirectional Long Short-Term Memory Network Network

EEG (Electroencephalography) signals are complex time-series data containing rich temporal and frequency information. Due to the dynamic nature of EEG signals, traditional signal processing methods struggle to capture long-term dependencies and dynamic changes effectively. Bi-LSTM, with its bidirectional mechanism, can more accurately model the temporal information in EEG signals, improving the accuracy of emotion recognition.

In EEG signal recognition models, Bi-LSTM processes EEG signals through forward and backward LSTM layers to extract deep features from the signals. These features can be used for further classification or regression tasks, aiding in recognizing participants' brain activity patterns in different emotional states. By incorporating attention mechanisms, the Bi-LSTM model can further focus on important moments in the signal, enhancing the model's recognition capabilities.

In Bi-LSTM, the forward LSTM layer processes EEG samples from time index 1 to t , generating the forward hidden state sequence \vec{h}_t , while the backward LSTM layer processes EEG samples from time index $t + 1$ to the end, generating the backward hidden state output \overleftarrow{h}_t . This bidirectional processing mechanism allows Bi-LSTM to capture dynamic changes in EEG signals and more accurately model long-term dependencies in the signals.

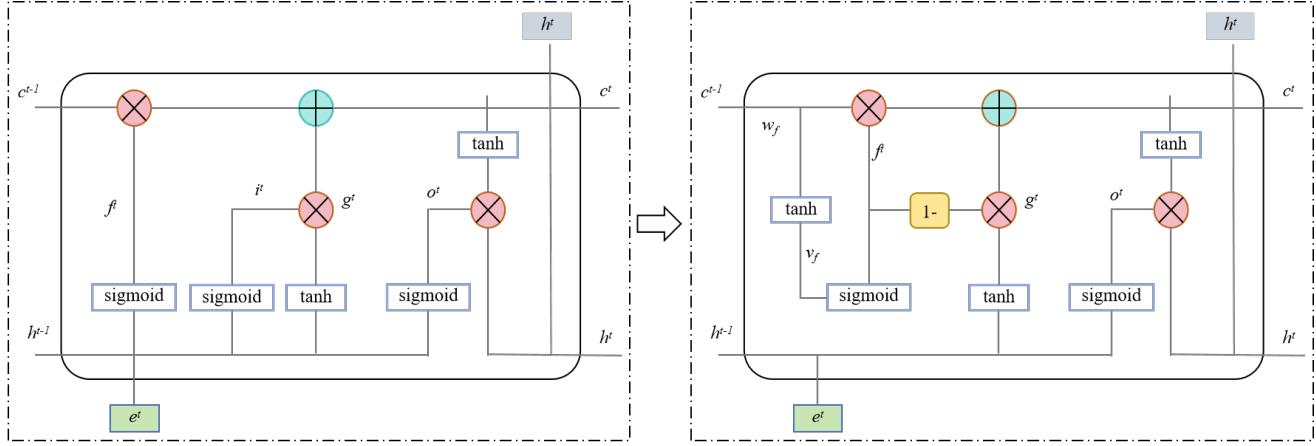


Fig. 4. Single LSTM Cell Diagram. Left: Original LSTM Cell Diagram; Right: LSTM Cell Diagram with Attention Gate.

The left part of Figure 4 illustrates the original architecture of LSTM, where the cell's output state update is related to the previous hidden layer output and the current input. At each time step t , the LSTM unit receives the current input x_t , the hidden state h_{t-1} , and the cell state c_{t-1} from the previous time step. The LSTM controls information storage and transfer through three gating mechanisms: the input gate, the forget gate, and the output gate. The input gate i_t determines the update of the cell state by the current input, the forget gate f_t decides whether to retain the previous cell state c_{t-1} , and the output gate o_t decides whether to transfer the hidden state h_{t-1} to the next LSTM unit. The candidate cell state \tilde{c}_t is a nonlinear transformation of the current input and the previous hidden state. The current cell state c_t is updated through the forget gate and the input gate, and the current hidden state h_t is determined by the output gate and the nonlinear transformation of the current cell state. Here, σ represents the sigmoid activation function, \tanh represents the tanh activation function, \odot denotes element-wise multiplication, W and U are weight matrices, and b is the bias vector. The formulas for the LSTM unit are shown below:

$$\begin{aligned} i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i), \\ f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f), \\ o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o), \\ \tilde{c}_t &= \tanh(W_c x_t + U_c h_{t-1} + b_c), \\ c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \\ h_t &= o_t \odot \tanh(c_t), \end{aligned} \quad (5)$$

Bi-LSTM combines the outputs of the forward and backward LSTM layers to form the final hidden state output h_t . The forward hidden state is computed by the forward LSTM layer, and the backward hidden state is computed by the backward LSTM layer. By combining the forward and backward hidden states, Bi-LSTM can more comprehensively understand the temporal information in EEG signals, leading to higher accuracy in emotion recognition. The formulas for the Bi-LSTM unit are shown below:

$$\begin{aligned} \vec{h}_t &= \text{LSTM}_\rightarrow(x_t, \overrightarrow{h_{t-1}}), \\ \overleftarrow{h}_t &= \text{LSTM}_\leftarrow(x_t, \overleftarrow{h_{t+1}}), \\ h_t &= [\vec{h}_t; \overleftarrow{h}_t], \end{aligned} \quad (6)$$

In emotion recognition models, Bi-LSTM effectively captures the temporal dynamics in EEG signals through its bidirectional mechanism. Using wearable frontal EEG signal monitoring devices, EEG signals of participants are recorded while they listen to different types of music. The device employs dry electrode technology, simplifying the signal acquisition process and making it suitable for large-scale daily applications. The collected EEG signals undergo preprocessing, including denoising, filtering, and feature extraction, to ensure signal quality and analysis accuracy. The Bi-LSTM model processes the preprocessed EEG signals through forward and backward LSTM layers, extracting deep features. By incorporating attention mechanisms to focus on important moments in the signals, this study introduces an improved LSTM method, as shown on the right side of Figure 4, which incorporates the attention mechanism to capture essential historical information and update the cell state. This ultimately improves the accuracy and robustness of emotion recognition, as represented by Eq. 7:

$$f_t = \delta(v_f * \tanh(w_f * c_{t-1})). \quad (7)$$

where v_f and w_f are the parameters of the attention mechanism. This method reduces the dimensionality of the training parameters compared to Eq. 5.

4. Experiment

4.1. Datasets

SEED dataset. The SEED dataset [74], collected by Shanghai Jiao Tong University, involves participants watching movie clips of various emotional types, with their EEG signals recorded in response to these stimuli. This dataset

Table 1
SEED dataset composition.

Number	Name of the clip	Label
1	Lost in Thailand	Vigorous
2	World Heritage in China	Neutral
3	Aftershock	Passive
4	Back to 1942	Passive
5	Flirting Scholar	Vigorous
6	Just Another Pandora's Box	Vigorous

was collected using a 62-channel EEG cap. The experiment included 15 participants, comprising 8 males and 7 females. Each participant was required to attend three identical sessions, with each session spaced 7-14 days apart. In each session, participants watched 15 movie clips, as detailed in Table 1. These 15 clips were selected from different segments of 6 movies, with each segment eliciting different emotions and affective states. During the experiment, the length of each clip was limited to approximately 4 minutes. After watching each clip, participants completed an emotional assessment and took a rest. Once the EEG data was acquired, it was downsampled and filtered for further analysis.

DEAP dataset. The DEAP dataset [75] is a multimodal dataset specifically designed for affective analysis, offering a rich collection of electroencephalogram (EEG), physiological signals, and video data. This dataset involves 32 participants who watched 40 one-minute-long music video excerpts, during which their emotional responses were recorded. The first part of the dataset comprises evaluations of the music videos by 14 to 16 participants, who rated the videos based on arousal, valence, and dominance. The second part of the dataset includes ratings, physiological recordings, and facial videos from 32 volunteers while watching the aforementioned 40 music video excerpts. In addition to emotional state ratings, the physiological recordings primarily consist of EEG data. The objective of the DEAP dataset is to provide researchers with a standardized dataset for testing and validating their methods of estimating emotional states, thereby advancing research in affective computing, emotion recognition, and brain-computer interfaces.

4.2. Data Preprocessing and Feature Extraction

EEG signals have low amplitudes and are often contaminated with various noise signals during acquisition. The preprocessing process aims to enhance the quality of EEG signals and reduce noise and artifacts. Common artifacts include electrooculographic (EOG) artifacts, electromyographic (EMG) artifacts, electrocardiographic (ECG) artifacts, skin conductance responses, and power line interference.

The EEG preprocessing workflow includes the following steps:

- (1) **Downsampling:** The original EEG signals were downsampled to reduce computational complexity (SEED: 1000 Hz to 200 Hz; DEAP: 512 Hz to 128 Hz).

Table 2
The rhythm of each band of EEG signal.

Band	Frequency (Hz)	Human State
Delta (δ)	0.1-3.0	Deep sleep, disordered, hypoxic, comatose states
Theta (θ)	4.0-7.0	Fatigue, depression, low mood, disappointment
Alpha (α)	8.0-12.0	Relaxed, calm, eyes closed but awake
Beta (β)	12.5-28.0	Tense, excited, happy
Gamma (γ)	29.0-50.0	Highly aroused, excited, tense

- (2) **Bad channel detection and removal:** Channels with excessive noise or artifacts were identified and removed.
- (3) **Electrode re-referencing:** Signals were re-referenced to a common average reference.
- (4) **Bandpass filtering:** A bandpass filter (0.5-50 Hz) was applied to remove irrelevant frequencies and noise.
- (5) **Artifact removal using ICA:** Independent Component Analysis (ICA) was performed to remove EOG, EMG, and other artifacts.

4.3. Emotion Evoked EEG Signal Generation and Acquisition

EEG is a method of recording the electrophysiological activity of the brain's neural tissues on the surface of the cerebral cortex. Neuronal excitation and inhibition in the brain generate voltage fluctuations, typically measured from the scalp in the range of 10 to 100 μ V in adults. Information in EEG signals is primarily contained within the frequency spectrum of 0.5 to several tens of Hertz. Based on frequency differences, EEG signals can be classified into five bands: δ band (Delta, 1-4 Hz), θ band (Theta, 4-8 Hz), α band (Alpha, 8-12 Hz), β band (Beta, 12-30 Hz), and γ band (Gamma, > 30 Hz). The rhythmic characteristics of each band are detailed in Table 2.

EEG signal collection generally employs either dry electrode or wet electrode methods. The current predominant method for EEG data collection is the wet electrode method, which has the advantage of obtaining more distinct EEG data but is inconvenient and less suitable for practical, everyday collection. The use of dry electrodes for EEG signal collection does not require an electrolyte, allowing the electrodes to contact the scalp directly, thereby offering greater convenience. However, due to the higher impedance of the stratum corneum, the EEG signals collected in this manner tend to be weaker. Figure 5 illustrates the layout of the 130-electrode system used in this study, which is both easy to implement and ensures test reproducibility. In the second phase of the experiment, 31 EEG channels were selected (indicated by the blue electrodes in Figure 5), with 13 channels located in the prefrontal cortex and 18 channels in the occipital lobe.

EEG signals have low amplitudes and are often contaminated with various noise signals during acquisition. The

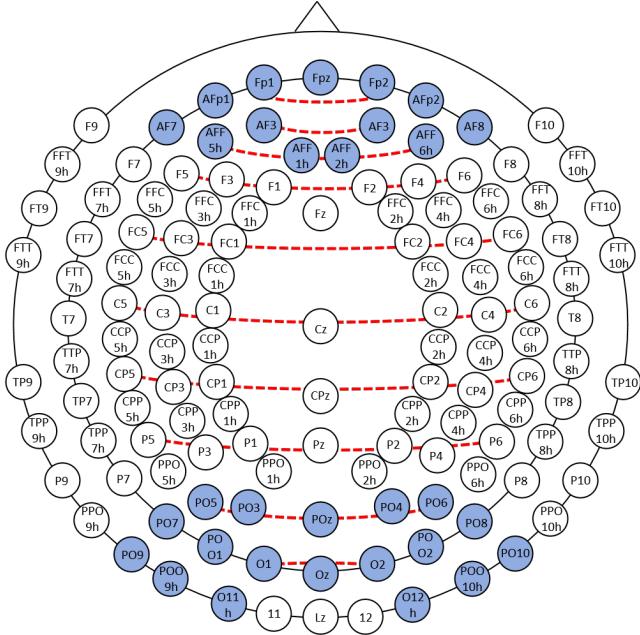


Fig. 5. EEG electrode 130 system placement method. The blue labels indicate the 31 EEG channels selected for the second phase of the experiment, with 13 channels in the prefrontal cortex and 18 channels in the occipital lobe.

preprocessing process aims to enhance the quality of EEG signals and reduce noise and artifacts. Common artifacts include electrooculographic (EOG) artifacts, electromyographic (EMG) artifacts, electrocardiographic (ECG) artifacts, skin conductance responses, and power line interference.

EOG artifacts typically occur in the frequency range of 0-15 Hz and are common EEG artifacts usually associated with eye movements. Blinking artifacts are characterized by narrow spikes with large amplitudes. EMG artifacts generally have higher frequencies and are mixed into EEG signals due to muscle movements during EEG experiments. ECG artifacts primarily arise from the signals generated by myocardial contraction and relaxation, typically presenting as low-frequency signals (less than 5 Hz). Skin conductance responses usually appear in signals collected from the palms or fingertips. Power line interference originates from the 50 Hz mains electricity voltage.

The EEG preprocessing workflow includes downsampling, bad channel detection, electrode re-referencing, bandpass filtering, and Independent Component Analysis (ICA) [76] for artifact removal.

4.4. Bi-LSTM Construction and Simulation Parameters

The Bi-LSTM model employed in this study is designed to process EEG signals effectively, leveraging both past and future information to capture long-term dependencies and enhance emotion recognition accuracy. Below are the details of the model construction, hyperparameters, time settings, and experimental parameters:

Bi-LSTM Model Architecture. The Bi-LSTM architecture consists of the following components: An enhanced Bi-LSTM layer with 128 hidden units in each direction (forward and backward); An attention weighting layer to emphasize critical features in the EEG signals; Two fully connected layers with 64 neurons each, using ReLU activation functions; A Softmax classification layer to output the probability distribution of emotional states.

Hyperparameters. The hyperparameters for the Bi-LSTM model were carefully selected based on empirical experiments and existing literature. The following settings were used: Learning rate: 0.001; Batch size: 64; Epochs: 100; Dropout rate: 0.5 to prevent overfitting; Optimizer: Adam.

Time Settings and Data Preprocessing. For the SEED dataset, EEG signals were recorded at a sampling rate of 1000 Hz and then downsampled to 200 Hz to reduce computational complexity. Each recording session lasted approximately 4 minutes per clip. The EEG signals were segmented into 1-second non-overlapping windows, resulting in 200 data points per segment.

For the DEAP dataset, EEG signals were recorded at a sampling rate of 512 Hz and then downsampled to 128 Hz. Each music video excerpt lasted 1 minute, and the signals were segmented into 1-second windows with 128 data points per segment.

Experimental Parameters and Settings. The experimental setup included the following key parameters: Feature extraction: Differential Entropy (DE) features were extracted from the segmented EEG signals; Model training: The Bi-LSTM model was trained on 80% of the data and validated on the remaining 20%; Evaluation metrics: Accuracy, precision, recall, and F1-score were used to evaluate the model's performance.

Simulation Process. The simulation of EEG signal fluctuations was conducted as follows: The Data_preprocessed files from the DEAP dataset were fed into a linear filter to map the spectral output of the EEG signals, representing brain neural activity; A receptive field model of the linear filter was created to generate artificial neural responses; The data were downsampled along the time dimension, and the receptive field was used to create an artificial neural response by performing a dot product with the receptive field; The Bi-LSTM model was then used to simulate and predict the time-varying stimulus response of the EEG signals.

4.5. Results

To ensure a fair comparison between the different models, we maintained consistent experimental conditions across all tests. All models were trained and evaluated using the same datasets (SEED and DEAP) and preprocessing techniques. Additionally, we used the same training, validation, and testing splits to provide an equitable basis for performance evaluation. Hyperparameters for each model were tuned individually to achieve optimal performance, ensuring that each model was fairly optimized for comparison. We also employed standard evaluation metrics, such as accuracy, precision, recall, and F1-score, to provide a

Table 3

Comparative Performance Analysis of Advanced Models on the SEED Dataset for EEG-Based Emotion Recognition.

Model	Multiclass	Channel	Accuracy(%)
SVM [77]	Binary	256	82.52
EEG-Net [79]	Binary	14	88.04
PMK [78]	Multi(3)	32	91.45
LSTM [80]	Multi(20)	64	86.75
Bi-LSTM	Multi(20)	64	92.39
Bi-LSTM+AttW(1 layer and 64 hidden size)	Multi(20)	64	93.42
Bi-LSTM+AttG(1 layer and 64 hidden size)	Multi(20)	64	95.16
Bi-LSTM+AttWG(1 layer and 32 hidden size)	Multi(20)	64	95.08
Bi-LSTM+AttWG(1 layer and 64 hidden size)	Multi(20)	128	98.28
Bi-LSTM+AttWG(1 layer and 128 hidden size)	Multi(20)	64	96.22

comprehensive assessment of each model's performance. The results demonstrate the superior performance of the Bi-LSTM model with attention mechanisms, highlighting the effectiveness of our proposed approach in emotion recognition using EEG signals.

Results on SEED dataset. We conducted a comparative experiment using EEG-based object detection visual models on the SEED dataset. The models we employed include Support Vector Machine (SVM) [77], Pyramid Match Kernel (PMK) [78], EEG-Net [79], LSTM [80], and Bi-LSTM. SVM is a supervised learning model used for classification and regression analysis, effectively handling high-dimensional data. PMK is a kernel function used for image matching by processing image features through a pyramid structure to achieve efficient matching. EEG-Net is a deep learning model specifically designed for processing and analyzing EEG data to improve the classification and recognition performance of EEG signals. LSTM is a type of RNN model that addresses long-term dependency issues through its gating mechanism, making it suitable for handling sequential data. Bi-LSTM consists of forward and backward LSTM layers, enabling the model to utilize both past and future information in the sequence, thus enhancing its understanding of temporal data.

Our proposal and the performance comparison of these models are shown in Table 3. In addition to these advanced model algorithms, we also compared our model with the incorporation of attention gates and attention weights to explore the effectiveness of Bi-LSTM. As shown in the table, the Bi-LSTM models with attention mechanisms significantly improved accuracy. Specifically, the Bi-LSTM-AttGW (1 layer and 128 hidden size) model achieved the highest accuracy of 98.28% in multiclass tasks. This demonstrates that incorporating attention mechanisms can significantly enhance the performance of models in emotion recognition using EEG signals. This method greatly improves the accuracy of capturing EEG signals.

As shown in Table 4, we compared the accuracy and standard deviation of different models in various frequency bands of the SEED dataset. Our proposed model demonstrated superior performance across all frequency bands, particularly in the γ band, where it achieved an accuracy of 98.28% with a standard deviation of 7.42. This highlights

Table 4

Comparison of Accuracy and Standard Deviation of Different Models in Different Frequency Bands of SEED Dataset.

Frequency Band	SVM [77]	LSTM [80]	Ours
δ	60.12 ± 14.11	74.45 ± 11.43	70.26 ± 15.36
θ	60.92 ± 10.25	71.26 ± 5.46	75.34 ± 15.16
α	$66.95 \text{ pm } 14.85$	74.31 ± 12.58	86.13 ± 14.83
β	80.71 ± 11.16	81.25 ± 10.11	90.23 ± 9.13
γ	82.52 ± 9.17	86.75 ± 8.43	98.28 ± 7.42

the robustness and effectiveness of our Bi-LSTM model with attention mechanisms in capturing the complexities of EEG signals and improving emotion recognition accuracy.

To visualize the EEG signal activity of the brain, we applied a low-pass filter at 8 Hz to the EEG signals from the SEED dataset to remove high-frequency noise. Figure 6 shows the EEG signal fragments in the regions of interest (ROIs) using the 130-electrode system layout. Each row in the figure represents the brain activity recorded for a segment from different movies, illustrating the variations in EEG signals in response to different emotional stimuli.

The first row represents the brain activity recorded for a segment from the movie *Lost in Thailand*, showing EEG signal snippets from 50 ms before to 130 ms after each beat mark. We performed baseline correction for each EEG signal band by subtracting the mean value calculated from the 50 ms segments between beat marks. The second row represents brain activity for a segment from the movie *Back to 1942*, recorded from 100 ms before to 180 ms after each beat mark, with baseline correction applied every 20 ms. The third row shows brain activity for a segment from the movie *World Heritage in China*, recorded from 55 ms before to 130 ms after each beat mark, with baseline correction applied every 25 ms. This detailed breakdown demonstrates the dynamic changes in brain activity elicited by different emotional stimuli across multiple time points.

When recording EEG data, the signal initially shows a negative dip at 0 ms. Any auditory processing related to the music beats occurs shortly afterward. One possible explanation is that the sudden change in the music beat causes a sudden dip in the EEG signal. However, this requires further

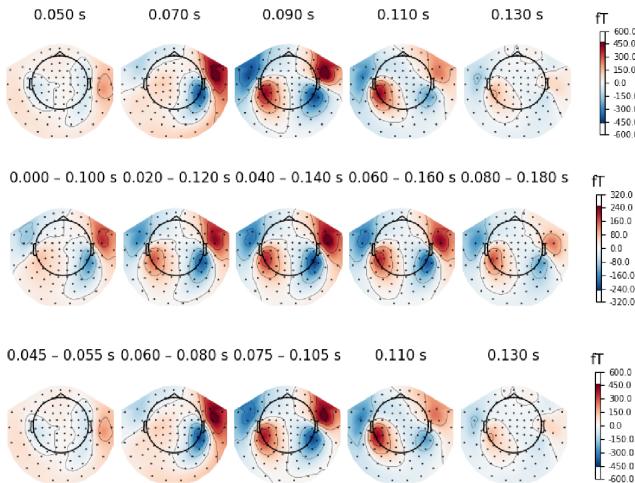


Fig. 6. Dynamic Changes in EEG Signal Regions of Interest at Different Time Points Across Various Movie Scenes in the SEED Datazset. The figure illustrates how different emotional stimuli from movies affect brain activity recorded through EEG signals.

investigation. It could also be due to inaccuracies in the wearable device used to collect the signal or the sampling rate and tracker precision. For further comparison, Figure 7 shows the EEG signal changes at the imagined time points corresponding to different movie scenes in the SEED dataset. This figure corresponds to the second row in Figure 6, showing the variations in the ROIs. The amplitude range of these EEG signal changes is relatively large, likely due to the emotional intensity provided by the movie segments. To calculate meaningful fluctuations, precise timing (e.g., sudden events) must be known. However, small changes during EEG signal recording are likely, causing some imprecision in the beat marks.

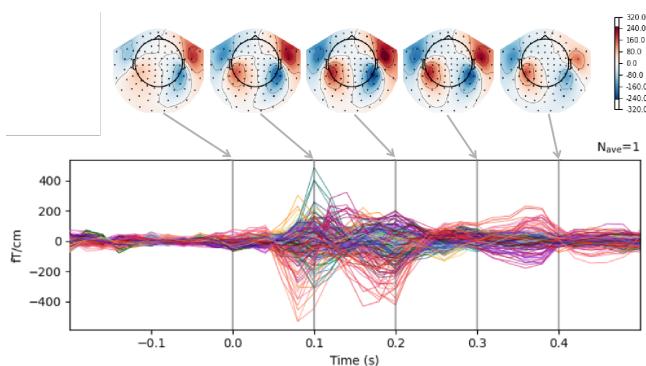


Fig. 7. EEG Signal Variations at Time Points in the Movie Scene "Back to 1942" from the SEED Dataset.

To understand the event-related effects on EEG signals induced by auditory and visual stimuli, we used the DEAP dataset, which contains emotion-related EEG signals. We analyzed the five EEG frequency bands affected by emotional states (as shown in Table 2). We employed Event-Related Potentials (ERP) metrics to record these changes.

Table 5
Statistics on the P300 value of ERP data on the DEAP dataset.

Emotion Change	ERP Change Period Post-Stimulus (ms)	P
Happiness (mean±SD)	298.45±33.25	0.026
Sadness (mean±SD)	316.45±30.24	0.470
Fear (mean±SD)	327.13±31.69	0.020

ERP refers to the specific potentials induced by stimuli such as auditory and visual cues, excluding spontaneous EEG signals. With its high temporal resolution, ERP can reflect changes in brain neurophysiological activities during cognitive processes, making it a widely used tool for assessing brain cognitive processing. The features commonly analyzed in ERP data are the latency and peaks of its components. The main components of ERP include P100, N100, P200, N200, and P300. Most studies focus on the P300 component of ERP data for brainwave analysis. The P300 component is a positive deflection occurring around 300 milliseconds post-stimulus. Table 5 presents the statistical results of P300 values from ERP data in the DEAP dataset. It is evident from the table that participants exhibit more intense emotional reactions to sad music scenes.

We used a Bi-LSTM model to simulate and predict EEG signal fluctuations recorded in the DEAP dataset. The Data_preprocessed files were fed into a linear filter to map the spectral output of the EEG signals, representing brain neural activity. We attempted to recreate the receptive field model of the linear filter to generate this data. The data were downsampled along the time dimension, and this receptive field was used to create an artificial neural response. This involved performing a dot product with the receptive field. As shown in Figure 8, the left side depicts the simulated neural response to EEG signals, with auditory features as input and the simulated time-varying stimulus response as output. The top right shows the receptive field generated by the stimulus signals, and the bottom right illustrates the EEG signal variation process simulated by the Bi-LSTM. The figure indicates that the EEG signals exhibit significant fluctuations in response to varying levels of musical stimuli, with the extent of fluctuations being related to the emotional intensity provided by the music.

5. Conclusion

This study records and analyzes brain activity while listening to music using wearable EEG devices and Bi-LSTM with attention mechanisms to identify emotional states. Data was collected using wearable EEG devices, preprocessed, and DE features were extracted. A Bi-LSTM model with attention mechanisms was then constructed, showing superior performance in emotion recognition, particularly with the SEED and DEAP datasets. The Bi-LSTM-AttGW model achieved the highest accuracy of 98.28% in multi-class tasks.

The novelty of this study lies in the integration of Bi-LSTM with attention mechanisms for EEG signal processing. This combination significantly improves the accuracy

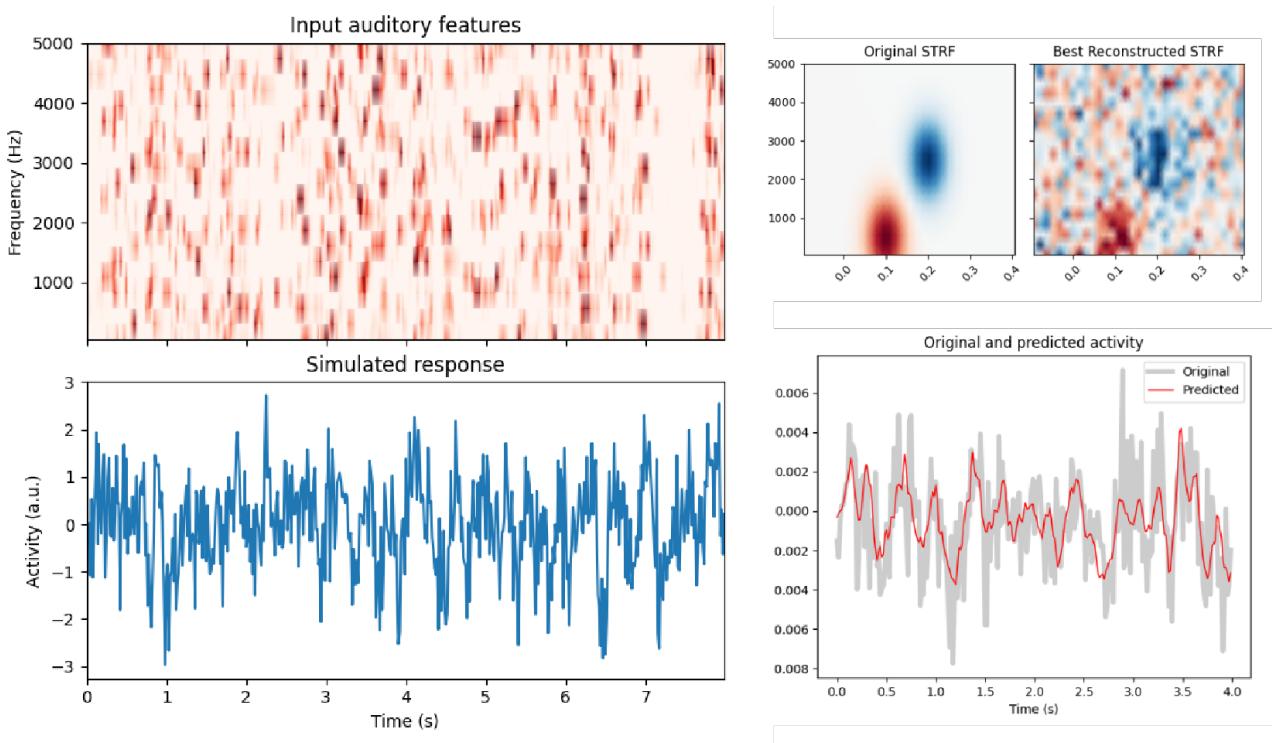


Fig. 8. Neural Response Signal Changes and Bi-LSTM Simulated Signal Variation Trends in EEG Data.

of emotion recognition, outperforming traditional models like SVM, EEG-Net, and LSTM [81, 82]. The study demonstrates the potential of using advanced deep learning techniques to better capture the complexities of EEG signals and enhance emotion recognition.

Despite these achievements, there are limitations. Wearable EEG devices have relatively low sampling rates and are prone to artifacts, affecting signal quality. Future research should focus on improving device design and signal processing algorithms. Additionally, larger sample sizes and more diverse experimental conditions are needed to account for individual differences in emotional responses to music. Further integration of multimodal data and various emotional elicitation methods could enhance the accuracy and applicability of emotion recognition in BCI and affective computing.

Author contributions

Jingyi Wang: Data Management, Methodology, Software, and Writing – original draft. **Zhiqun Wang:** Supervision, Writing – review & editing. Fanyu Kong: Methodology, Software, Writing – review & editing. **Guiran Liu:** Supervision, 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.

Data availability

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

- [1] X. Hu, S. Yuan, F. Xu, Y. Leng, K. Yuan, Q. Yuan, Scalp eeg classification using deep bi-lstm network for seizure detection, *Computers in Biology and Medicine* 124 (2020) 103919.
- [2] Y. Wang, S. Sun, Q. Guo, The mechanism of the impact of enterprise digital transformation on transaction performance, *Journal of Xi'an University of Finance and Economics* 37 (2024) 60–71.
- [3] Y. Wang, W. Song, W. Tao, A. Liotta, D. Yang, X. Li, S. Gao, Y. Sun, W. Ge, W. Zhang, et al., A systematic review on affective computing: Emotion models, databases, and recent advances, *Information Fusion* 83 (2022) 19–52.
- [4] K. S. Kamble, J. Sengupta, Ensemble machine learning-based affective computing for emotion recognition using dual-decomposed eeg signals, *IEEE Sensors Journal* 22 (2021) 2496–2507.
- [5] D. S. Naser, G. Saha, Influence of music liking on eeg based emotion recognition, *Biomedical Signal Processing and Control* 64 (2021) 102251.
- [6] I. Daly, Neural decoding of music from the eeg, *Scientific Reports* 13 (2023) 624.
- [7] M. M. Rahman, A. K. Sarkar, M. A. Hossain, M. S. Hossain, M. R. Islam, M. B. Hossain, J. M. Quinn, M. A. Moni, Recognition of human emotions using eeg signals: A review, *Computers in biology and medicine* 136 (2021) 104696.
- [8] X. Zheng, W. Chen, An attention-based bi-lstm method for visual object classification via eeg, *Biomedical Signal Processing and Control* 63 (2021) 102174.
- [9] S. Dai, K. Li, Z. Luo, P. Zhao, B. Hong, A. Zhu, J. Liu, Ai-based nlp section discusses the application and effect of bag-of-words models

- and tf-idf in nlp tasks, *Journal of Artificial Intelligence General science (JAIGS) ISSN: 3006-4023 5* (2024) 13–21.
- [10] Y. Wang, M. Alangari, J. Hihath, A. K. Das, M. Anantram, A machine learning approach for accurate and real-time dna sequence identification, *BMC genomics* 22 (2021) 1–10.
- [11] A. Richardson, X. Wang, A. Dubey, J. Sprinkle, Reinforcement learning with communication latency with application to stop-and-go wave dissipation, in: *2024 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2024, pp. 1187–1193.
- [12] G. Bouallegue, R. Djemal, S. A. Alshebeili, H. Aldhalaan, A dynamic filtering df-rnn deep-learning-based approach for eeg-based neurological disorders diagnosis, *IEEE Access* 8 (2020) 206992–207007.
- [13] J. Wang, F. Li, Y. An, X. Zhang, H. Sun, Towards robust lidar-camera fusion in bev space via mutual deformable attention and temporal aggregation, *IEEE Transactions on Circuits and Systems for Video Technology* (2024) 1–1.
- [14] Z. Xu, D. Deng, Y. Dong, K. Shimada, Dpmpc-planner: A real-time uav trajectory planning framework for complex static environments with dynamic obstacles, in: *2022 International Conference on Robotics and Automation (ICRA)*, IEEE, 2022, pp. 250–256.
- [15] P. Zhao, K. Li, B. Hong, A. Zhu, J. Liu, S. Dai, Task allocation planning based on hierarchical task network for national economic mobilization, *Journal of Artificial Intelligence General science (JAIGS) ISSN: 3006-4023 5* (2024) 22–31.
- [16] Y. Song, R. Fellegara, F. Iuricich, L. De Floriani, Parallel topology-aware mesh simplification on terrain trees, *ACM Transactions on Spatial Algorithms and Systems* 10 (2024) 1–39.
- [17] X. Geng, X. Zhang, M. Yue, W. Hu, L. Wang, X. Zhang, P. Yu, D. Long, H. Yan, A motor imagery eeg signal optimized processing algorithm, *Alexandria Engineering Journal* 101 (2024) 38–51.
- [18] Q. Zhang, W. Qi, H. Zheng, X. Shen, Cu-net: a u-net architecture for efficient brain-tumor segmentation on brats 2019 dataset, 2024, arXiv:2406.13113.
- [19] H. Peng, R. Ran, Y. Luo, J. Zhao, S. Huang, K. Thorat, T. Geng, C. Wang, X. Xu, W. Wen, et al., Lingcn: Structural linearized graph convolutional network for homomorphically encrypted inference, in: *Thirty-seventh Conference on Neural Information Processing Systems*, ????
- [20] Y. Song, R. Fellegara, F. Iuricich, L. De Floriani, Efficient topology-aware simplification of large triangulated terrains, in: *Proceedings of the 29th International Conference on Advances in Geographic Information Systems*, 2021, pp. 576–587.
- [21] Y. Wang, V. Khandelwal, A. K. Das, M. Anantram, Classification of dna sequences: Performance evaluation of multiple machine learning methods, in: *2022 IEEE 22nd International Conference on Nanotechnology (NANO)*, IEEE, 2022, pp. 333–336.
- [22] R. Zanetti, A. Arza, A. Aminifar, D. Atienza, Real-time eeg-based cognitive workload monitoring on wearable devices, *IEEE transactions on biomedical engineering* 69 (2021) 265–277.
- [23] P. Srinivas, M. Arulprakash, M. Vadivel, N. Anusha, G. Rajasekar, C. Srinivasan, Support vector machines based predictive seizure care using iot-wearable eeg devices for proactive intervention in epilepsy, in: *2024 2nd International Conference on Computer, Communication and Control (IC4)*, IEEE, 2024, pp. 1–5.
- [24] K. Kotowski, K. Stapor, J. Leski, M. Kotas, Validation of emotiv epoc+ for extracting erp correlates of emotional face processing, *Biocybernetics and Biomedical Engineering* 38 (2018) 773–781.
- [25] Q. Cheng, Y. Song, The impact of the digital economy on regional economic development disparities from the perspective of spatial spillovers, *Journal of Xi'an University of Finance and Economics* 36 (2023) 44–57.
- [26] J. H. Shin, J. Kwon, J. U. Kim, H. Ryu, J. Ok, S. Joon Kwon, H. Park, T.-i. Kim, Wearable eeg electronics for a brain-ai closed-loop system to enhance autonomous machine decision-making, *npj Flexible Electronics* 6 (2022) 32.
- [27] T. Li, R. Zhao, Y. Liu, X. Liu, Y. Li, Effect of age on driving behavior and a neurophysiological interpretation, in: *International Conference on Human-Computer Interaction*, Springer, 2022, pp. 184–194.
- [28] A. De, H. Mohammad, Y. Wang, R. Kubendran, A. K. Das, M. Anantram, Modeling and simulation of dna origami based electronic read-only memory, in: *2022 IEEE 22nd International Conference on Nanotechnology (NANO)*, IEEE, 2022, pp. 385–388.
- [29] B. Hong, P. Zhao, J. Liu, A. Zhu, S. Dai, K. Li, The application of artificial intelligence technology in assembly techniques within the industrial sector, *Journal of Artificial Intelligence General science (JAIGS) ISSN: 3006-4023 5* (2024) 1–12.
- [30] C. Jin, T. Huang, Y. Zhang, M. Pechenizkiy, S. Liu, S. Liu, T. Chen, Visual prompting upgrades neural network sparsification: A data-model perspective, arXiv preprint arXiv:2312.01397 (2023).
- [31] Y. Dong, The design of autonomous uav prototypes for inspecting tunnel construction environment, arXiv preprint arXiv:2408.07286 (2024).
- [32] R. Zhao, Y. Liu, T. Li, Y. Li, A preliminary evaluation of driver's workload in partially automated vehicles, in: *International Conference on Human-Computer Interaction*, Springer, 2022, pp. 448–458.
- [33] C. Jin, T. Che, H. Peng, Y. Li, M. Pavone, Learning from teaching regularization: Generalizable correlations should be easy to imitate, arXiv preprint arXiv:2402.02769 (2024).
- [34] X. Jiang, J. Yu, Z. Qin, Y. Zhuang, X. Zhang, Y. Hu, Q. Wu, Dualvd: An adaptive dual encoding model for deep visual understanding in visual dialogue, in: *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 2020, pp. 11125–11132.
- [35] T. Li, R. Zhao, Y. Liu, Y. Li, G. Li, Evaluate the effect of age and driving experience on driving performance with automated vehicles, in: *International Conference on Applied Human Factors and Ergonomics*, Springer, 2021, pp. 155–161.
- [36] X. Xie, H. Peng, A. Hasan, S. Huang, J. Zhao, H. Fang, W. Zhang, T. Geng, O. Khan, C. Ding, Accel-gcn: High-performance gpu accelerator design for graph convolution networks, in: *2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD)*, IEEE, 2023, pp. 01–09.
- [37] P. Chen, Z. Zhang, Y. Dong, L. Zhou, H. Wang, Enhancing visual question answering through ranking-based hybrid training and multi-modal fusion, arXiv preprint arXiv:2408.07303 (2024).
- [38] K. Li, A. Zhu, W. Zhou, P. Zhao, J. Song, J. Liu, Utilizing deep learning to optimize software development processes, arXiv preprint arXiv:2404.13630 (2024).
- [39] V. Lambrecq, A. Hanin, E. Munoz-Musat, L. Chougar, S. Gassama, C. Delorme, L. Cousyn, A. Borden, M. Damiano, V. Frazzini, et al., Association of clinical, biological, and brain magnetic resonance imaging findings with electroencephalographic findings for patients with covid-19, *JAMA Network Open* 4 (2021) e211489–e211489.
- [40] M. Nentwich, L. Ai, J. Madsen, Q. K. Telesford, S. Haufe, M. P. Milham, L. C. Parra, Functional connectivity of eeg is subject-specific, associated with phenotype, and different from fmri, *NeuroImage* 218 (2020) 117001.
- [41] J. M. Palva, S. Palva, Infra-slow fluctuations in electrophysiological recordings, blood-oxygenation-level-dependent signals, and psychophysical time series, *Neuroimage* 62 (2012) 2201–2211.
- [42] E. A. Wilde, N. J. Goodrich-Hunsaker, A. L. Ware, B. A. Taylor, B. D. Bickman, J. V. Hunter, R. Newman-Norlund, S. Scarneo, D. J. Casa, H. S. Levin, Diffusion tensor imaging indicators of white matter injury are correlated with a multimodal electroencephalography-based biomarker in slow recovering, concussed collegiate athletes, *Journal of neurotrauma* 37 (2020) 2093–2101.
- [43] S. Tortora, S. Ghidoni, C. Chisari, S. Micera, F. Artoni, Deep learning-based bci for gait decoding from eeg with lstm recurrent neural network, *Journal of neural engineering* 17 (2020) 046011.
- [44] S. J. Teipel, O. Pogarell, T. Meindl, O. Dietrich, D. Sydykova, U. Hunklinger, B. Georgii, C. Muler, M. F. Reiser, H.-J. Möller, et al., Regional networks underlying interhemispheric connectivity: an eeg and dti study in healthy ageing and amnestic mild cognitive impairment, *Human brain mapping* 30 (2009) 2098–2119.
- [45] H. Cui, A. Liu, X. Zhang, X. Chen, J. Liu, X. Chen, Eeg-based subject-independent emotion recognition using gated recurrent unit

- and minimum class confusion, *IEEE Transactions on Affective Computing* 14 (2023) 2740–2750.
- [46] T. Li, G. Pang, X. Bai, J. Zheng, L. Zhou, X. Ning, Learning adversarial semantic embeddings for zero-shot recognition in open worlds, *Pattern Recognition* 149 (2024) 110258.
- [47] H. Ran, W. Li, L. Li, S. Tian, X. Ning, P. Tiwari, Learning optimal inter-class margin adaptively for few-shot class-incremental learning via neural collapse-based meta-learning, *Information Processing & Management* 61 (2024) 103664.
- [48] A. A. Ein Shoka, M. M. Dessouky, A. El-Sayed, E. El-Din Hemdan, An efficient cnn based epileptic seizures detection framework using encrypted eeg signals for secure telemedicine applications, *Alexandria Engineering Journal* 65 (2023) 399–412.
- [49] A. Zhu, K. Li, T. Wu, P. Zhao, W. Zhou, B. Hong, Cross-task multi-branch vision transformer for facial expression and mask wearing classification, *arXiv preprint arXiv:2404.14606* (2024).
- [50] K. Li, P. Xirui, J. Song, B. Hong, J. Wang, The application of augmented reality (ar) in remote work and education, *arXiv preprint arXiv:2404.10579* (2024).
- [51] Z. An, X. Wang, T. T. Johnson, J. Sprinkle, M. Ma, Runtime monitoring of accidents in driving recordings with multi-type logic in empirical models, in: *International Conference on Runtime Verification*, Springer, 2023, pp. 376–388.
- [52] Y. Liu, R. Zhao, T. Li, Y. Li, An investigation of the impact of autonomous driving on driving behavior in traffic jam, in: *IIE Annual Conference. Proceedings*, Institute of Industrial and Systems Engineers (IISE), 2021, pp. 986–991.
- [53] S. Sheykhan, Z. Mousavi, T. Y. Rezaii, A. Farzamnia, Recognizing emotions evoked by music using cnn-lstm networks on eeg signals, *IEEE access* 8 (2020) 139332–139345.
- [54] X. Wang, S. Onwumelu, J. Sprinkle, Using automated vehicle data as a fitness tracker for sustainability, in: *2024 Forum for Innovative Sustainable Transportation Systems (FISTS)*, IEEE, 2024, pp. 1–6.
- [55] R. Fellegara, F. Iuricich, Y. Song, L. D. Floriani, Terrain trees: a framework for representing, analyzing and visualizing triangulated terrains, *GeoInformatica* 27 (2023) 525–564.
- [56] H. Peng, S. Huang, T. Zhou, Y. Luo, C. Wang, Z. Wang, J. Zhao, X. Xie, A. Li, T. Geng, et al., Autorep: Automatic relu replacement for fast private network inference, in: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, 2023, pp. 5155–5165.
- [57] Y. Zhuang, Y. Chen, J. Zheng, Music genre classification with transformer classifier, in: *Proceedings of the 2020 4th international conference on digital signal processing*, 2020, pp. 155–159.
- [58] R. Zhao, Y. Liu, Y. Li, B. Tokgoz, An investigation of resilience in manual driving and automatic driving in freight transportation system, in: *IIE Annual Conference. Proceedings*, Institute of Industrial and Systems Engineers (IISE), 2021, pp. 974–979.
- [59] A. De, H. Mohammad, Y. Wang, R. Kubendran, A. K. Das, M. Anantram, Performance analysis of dna crossbar arrays for high-density memory storage applications, *Scientific Reports* 13 (2023) 6650.
- [60] Y. Liu, R. Zhao, Y. Li, A preliminary comparison of drivers' overtaking behavior between partially automated vehicles and conventional vehicles, in: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 66, SAGE Publications Sage CA: Los Angeles, CA, 2022, pp. 913–917.
- [61] B. Somers, C. J. Long, T. Francart, Eeg-based diagnostics of the auditory system using cochlear implant electrodes as sensors, *Scientific Reports* 11 (2021) 5383.
- [62] D. Henao, M. Navarrete, M. Valderrama, M. Le Van Quyen, Entrainment and synchronization of brain oscillations to auditory stimulations, *Neuroscience Research* 156 (2020) 271–278.
- [63] M. L. Ferster, G. Da Poian, K. Menachery, S. J. Schreiner, C. Lustenberger, A. Maric, R. Huber, C. R. Baumann, W. Karlen, Benchmarking real-time algorithms for in-phase auditory stimulation of low amplitude slow waves with wearable eeg devices during sleep, *IEEE Transactions on Biomedical Engineering* 69 (2022) 2916–2925.
- [64] X. Geng, D. Li, H. Chen, P. Yu, H. Yan, M. Yue, An improved feature extraction algorithms of eeg signals based on motor imagery brain-computer interface, *Alexandria Engineering Journal* 61 (2022) 4807–4820.
- [65] Q. Deng, Z. Fan, Z. Li, X. Pan, Q. Kang, M. Zhou, Solving the food-energy-water nexus problem via intelligent optimization algorithms, *arXiv preprint arXiv:2404.06769* (2024).
- [66] S. Patel, Y. Liu, R. Zhao, X. Liu, Y. Li, Inspection of in-vehicle touchscreen infotainment display for different screen locations, menu types, and positions, in: *International conference on human-computer interaction*, Springer, 2022, pp. 258–279.
- [67] J. Lee, H. Wang, K. Jang, A. Hayat, M. Bunting, A. Alanqary, W. Barbour, Z. Fu, X. Gong, G. Gunter, et al., Traffic smoothing via connected & automated vehicles: A modular, hierarchical control design deployed in a 100-cav flow smoothing experiment, *IEEE Control Systems Magazine* (2024).
- [68] Z. Zhu, R. Zhao, J. Ni, J. Zhang, Image and spectrum based deep feature analysis for particle matter estimation with weather informatio, in: *2019 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2019, pp. 3427–3431.
- [69] Y. Liu, R. Zhao, T. Li, Y. Li, The impact of directional road signs combinations and language unfamiliarity on driving behavior, in: *International Conference on Human-Computer Interaction*, Springer, 2022, pp. 195–204.
- [70] J. Zhang, Z. Wei, J. Zou, H. Fu, Automatic epileptic eeg classification based on differential entropy and attention model, *Engineering Applications of Artificial Intelligence* 96 (2020) 103975.
- [71] T. Zhou, J. Zhao, Y. Luo, X. Xie, W. Wen, C. Ding, X. Xu, Adapi: Facilitating dnn model adaptivity for efficient private inference in edge computing, *arXiv preprint arXiv:2407.05633* (2024).
- [72] H. Peng, X. Xie, K. Shivdikar, M. A. Hasan, J. Zhao, S. Huang, O. Khan, D. Kaeli, C. Ding, Maxk-gnn: Extremely fast gpu kernel design for accelerating graph neural networks training, in: *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, Volume 2, ASPLOS '24, Association for Computing Machinery, New York, NY, USA, 2024, p. 683–698.
- [73] C. Jin, H. Peng, S. Zhao, Z. Wang, W. Xu, L. Han, J. Zhao, K. Zhong, S. Rajasekaran, D. N. Metaxas, Apeer: Automatic prompt engineering enhances large language model reranking, *arXiv preprint arXiv:2406.14449* (2024).
- [74] Y. Cimtay, E. Ekmekcioglu, Investigating the use of pretrained convolutional neural network on cross-subject and cross-dataset eeg emotion recognition, *Sensors* 20 (2020) 2034.
- [75] M. Khateeb, S. M. Anwar, M. Alnowami, Multi-domain feature fusion for emotion classification using deap dataset, *Ieee Access* 9 (2021) 12134–12142.
- [76] M. J. Antony, B. P. Sankaralingam, R. K. Mahendran, A. A. Gardezi, M. Shafiq, J.-G. Choi, H. Hamam, Classification of eeg using adaptive svm classifier with csp and online recursive independent component analysis, *Sensors* 22 (2022) 7596.
- [77] J. Kang, X. Han, J. Song, Z. Niu, X. Li, The identification of children with autism spectrum disorder by svm approach on eeg and eye-tracking data, *Computers in biology and medicine* 120 (2020) 103722.
- [78] F. Hou, J. Liu, Z. Bai, Z. Yang, J. Liu, Q. Gao, Y. Song, Eeg-based emotion recognition for hearing impaired and normal individuals with residual feature pyramids network based on time-frequency-spatial features, *IEEE Transactions on Instrumentation and Measurement* 72 (2023) 1–11.
- [79] D. Huang, X. Wang, J. Liu, J. Li, W. Tang, Virtual reality safety training using deep eeg-net and physiology data, *The visual computer* 38 (2022) 1195–1207.
- [80] N. Kadri, A. Ellouze, M. Ksantini, S. H. Turki, New lstm deep learning algorithm for driving behavior classification, *Cybernetics and Systems* 54 (2023) 387–405.
- [81] Y. Wang, B. Demir, H. Mohammad, E. E. Oren, M. Anantram, Computational study of the role of counterions and solvent dielectric in determining the conductance of b-dna, *Physical Review E* 107

- (2023) 044404.
- [82] Y. Luo, N. Xu, H. Peng, C. Wang, S. Duan, K. Mahmood, W. Wen, C. Ding, X. Xu, Aq2pnn: Enabling two-party privacy-preserving deep neural network inference with adaptive quantization, in: 2023 56th IEEE/ACM International Symposium on Microarchitecture (MICRO), IEEE, 2023, pp. 628–640.

NEUROLM: A UNIVERSAL MULTI-TASK FOUNDATION MODEL FOR BRIDGING THE GAP BETWEEN LANGUAGE AND EEG SIGNALS

Wei-Bang Jiang^{1*}, Yansen Wang², Bao-Liang Lu¹, Dongsheng Li²

¹Shanghai Jiao Tong University ²Microsoft Research Asia

{935963004, bllu}@sjtu.edu.cn, {yansenwang, dongsli}@microsoft.com

ABSTRACT

Recent advancements for large-scale pre-training with neural signals such as electroencephalogram (EEG) have shown promising results, significantly boosting the development of brain-computer interfaces (BCIs) and healthcare. However, these pre-trained models often require full fine-tuning on each downstream task to achieve substantial improvements, limiting their versatility and usability, and leading to considerable resource wastage. To tackle these challenges, we propose NeuroLM, the first multi-task foundation model that leverages the capabilities of Large Language Models (LLMs) by regarding EEG signals as a foreign language, endowing the model with multi-task learning and inference capabilities. Our approach begins with learning a text-aligned neural tokenizer through vector-quantized temporal-frequency prediction, which encodes EEG signals into discrete neural tokens. These EEG tokens, generated by the frozen vector-quantized (VQ) encoder, are then fed into an LLM that learns causal EEG information via multi-channel autoregression. Consequently, NeuroLM can understand both EEG and language modalities. Finally, multi-task instruction tuning adapts NeuroLM to various downstream tasks. We are the first to demonstrate that, by specific incorporation with LLMs, NeuroLM unifies diverse EEG tasks within a single model through instruction tuning. The largest variant NeuroLM-XL has record-breaking 1.7B parameters for EEG signal processing, and is pre-trained on a large-scale corpus comprising approximately 25,000-hour EEG data. When evaluated on six diverse downstream datasets, NeuroLM showcases the huge potential of this multi-task learning paradigm.

1 INTRODUCTION

Electroencephalogram (EEG) signals have become a cornerstone in the development of brain-computer interfaces and healthcare domains, offering a non-invasive solution to capture the electrical activity of the brain. EEG measures the voltage fluctuations resulting from ionic current flows within the neurons of the brain, providing real-time insights into brain function and neural dynamics. This capability makes EEG an invaluable tool for creating interfaces that enable direct communication between the brain and external devices. EEG is advantageous due to its high temporal resolution, cost-effectiveness, and portability, and has been significantly enhanced by advanced computational methods. Therefore, a wide range of applications have been utilizing EEG signals, including but not limited to human emotion recognition (Jenke et al., 2014), body motor imagery (Tabar & Halici, 2016), automatic sleep stage classification (Supratak et al., 2017), seizure epilepsy detection (Alotaiby et al., 2014), and fatigue detection (Gao et al., 2019).

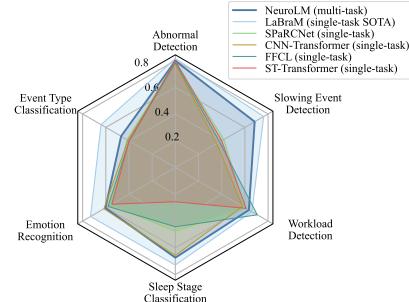


Figure 1: The overall performance comparison on six tasks.

*Work done during Wei-Bang’s internship at Microsoft Research Asia. Correspondence to Yansen Wang.

While EEG signals are popular among researchers, they have several disadvantages, including the low signal-to-noise ratio, inherent nonstationarity, as well as diverse configurations in EEG data collection. Besides, there is a lack of sufficient and consistent EEG data. These challenges complicate the extraction of universal EEG representations. To overcome these problems, several studies have proposed methods compatible with diverse EEG configurations to learn effective and generic representations. For example, Yang et al. (2023a) introduce a Biosignal Transformer (BIOT), which unifies various EEG data by tokenizing channels into fix-length segments with channel and relative position embeddings for preserving spatio-temporal features. Jiang et al. (2024) advance this approach by proposing a neural tokenizer to pre-train LaBraM by masked neural code prediction with 2,500 hours of EEG data, thus achieving state-of-the-art (SOTA) performance on various downstream tasks. Although these methods effectively address the aforementioned challenges, they still require individual fine-tuning on each downstream dataset to obtain impressive improvement. Despite increasing model size and employing large-scale unsupervised pre-training to learn generic representations, such adaptation confines the fine-tuned model to perform only a single task. Moreover, this task-specific fine-tuning demands substantial computational and storage resources.

Over the past few years, the advent of Large Language Models has brought remarkable progress and demonstrated extraordinary emergent abilities (Brown et al., 2020; Touvron et al., 2023). The development of LLMs has given rise to Multimodal Large Language Models (MLLMs) (Achiam et al., 2023; Liu et al., 2023), which unleash the potential of powerful LLMs to perform multimodal tasks. MLLMs typically integrate a modality-specific encoder, pre-aligned with text embeddings, into off-the-shelf LLMs. Inspired by MLLMs, we unveil a new direction of integrating multiple EEG tasks into a unified model by incorporating EEG signals into existing LLMs. However, there are some challenges in harnessing LLMs to understand EEG patterns, comprising:

- 1) EEG-text embedding alignment.** Aligning EEG and text embeddings presents a great challenge. Unlike vision-language models which benefit from numerous high-quality image-text pairs, there are no established EEG-text pairs available due to the difficulty of extracting semantic information from a given EEG segment.
- 2) Effective Representation learning with LLMs.** Mainstream methods employ masked EEG modeling to effectively extract representations for EEG signals. When integrating LLMs, how to learn generic information within the LLM paradigm remains an unsolved issue.
- 3) Unified multi-task learning with various EEG tasks.** Integrating multiple EEG tasks into a unified model is complex due to the diversity and specificity of different tasks. Developing a model that can seamlessly handle various tasks without compromising performance on any individual task is a major challenge.

In light of the aforementioned challenges, we propose NeuroLM, a universal multi-task foundation model for EEG signal processing. NeuroLM builds upon the compatibility with diverse EEG formats established by LaBraM, and it is pre-trained on a large-scale dataset comprising approximately 25,000 hours of EEG data. The training of NeuroLM involves three stages. First, a text-aligned neural tokenizer is trained using vector-quantized temporal-frequency prediction to encode continuous EEG signals into discrete codes from a neural codebook, with adversarial training employed to align the EEG and text spaces. Next, the VQ encoder of the neural tokenizer is frozen to extract compact embeddings, which serve as input for a LLM. To enable the LLM to learn causal EEG representations, we propose multi-channel autoregressive pre-training, which mimics autoregressive language modeling but is tailored for multi-channel EEG signals. Finally, we elaborate instructions for various downstream datasets and employ multi-task instruction tuning to empower NeuroLM for multi-task learning. Experiments on six different tasks, encompassing abnormal detection, event type classification, emotion recognition, sleep stage classification, cognitive workload prediction, and slowing type classification, demonstrate NeuroLM’s superiority in multi-task learning and inference. To the best of our knowledge, we are the first to introduce instruction tuning to enable multi-task learning and inference in the field of EEG signal processing. The highlights are summarized as follows:

- 1) Text-aligned neural tokenizer embeddings.** We introduce a text-aligned neural tokenizer that effectively bridges the gap between EEG and text data. This tokenizer uses vector-quantized temporal-frequency prediction to convert EEG signals into discrete codes, facilitating the alignment of EEG and text embeddings through adversarial training. This alignment is crucial for leveraging the strengths of LLMs in understanding and processing EEG data.

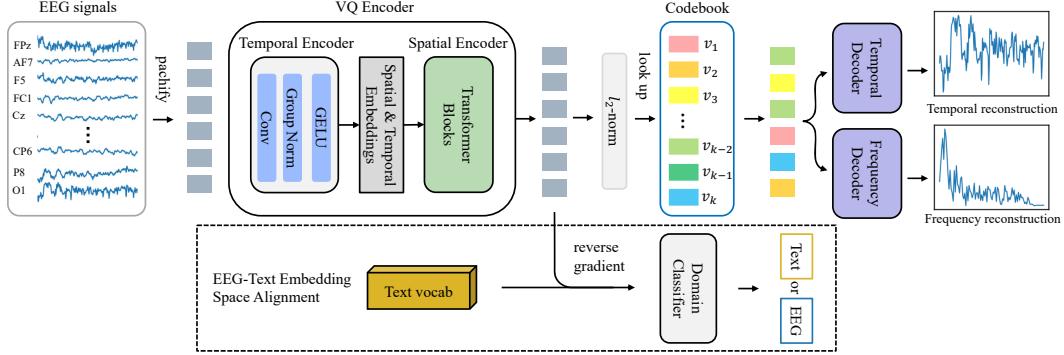


Figure 2: The architecture design of text-aligned neural tokenizer training. The neural tokenizer is trained by reconstructing both temporal and frequency domain of input EEG signals to discretize them into discrete neural tokens. To align EEG and text embedding space, we utilize a domain classifier through adversarial training.

2) Large-scale multi-channel autoregressive pre-training. NeuroLM employs multi-channel autoregression, enabling the model to learn causal representations across different EEG channels. Pre-training on 25,000 hours of EEG data ensures that NeuroLM captures a wide range of neural patterns, enhancing its ability to generalize across diverse EEG tasks.

3) Joint multi-task tuning and inference. We pioneer the use of joint multi-task tuning and inference for EEG. By elaborating specific instructions for various downstream tasks and employing multi-task instruction tuning, NeuroLM is capable of performing multiple tasks within a single model. This not only improves efficiency by reducing the need for individual fine-tuning for each task but also ensures high performance across a spectrum of applications.

2 METHOD

In this section, we elaborate our design of NeuroLM. We first train a neural tokenizer by vector-quantized temporal-frequency prediction. Whereafter, the VQ encoder of the tokenizer will serve to encode EEG signals into embeddings aligned with text space, and the EEG embeddings will be seamlessly used as input to Large Language Models.

Given multi-channel EEG signals $X \in \mathbb{R}^{C \times T}$, where C denotes the number of channels and T denotes total timestamps. An EEG sample is formulated as $x \in \mathbb{R}^{C \times L}$, where L is the window size, resulting in a total number of $\lfloor \frac{T}{L} \rfloor$ samples. We patchify the EEG samples into non-overlap patches $x = \{x_{ij} \in \mathbb{R}^P | i = 1, \dots, C, j = 1, \dots, N\}$. Let P is patch size and $N = \lfloor \frac{L}{P} \rfloor$.

2.1 TEXT-ALIGNED NEURAL TOKENIZER TRAINING

To incorporate EEG into off-the-shelf Large Language Models, we first need to encode EEG signals into embeddings whose space is well-aligned with text embedding space. VQ-VAE (Van Den Oord et al., 2017) is a good choice that maps continuous signals to discrete tokens while preserving the key information. Our text-aligned neural tokenizer basically follows the well-established neural tokenizer of LaBraM (Jiang et al., 2024) with some improvements. Vector-quantized temporal-frequency prediction is utilized to train the text-aligned neural tokenizer, as illustrated in Figure 2.

Neural Tokenizer. The neural tokenizer is composed of several vital components: VQ encoder, codebook, temporal/frequency decoder, and domain classifier. The codebook $V \in \mathbb{R}^{K \times D}$ contains K discrete D -dimension embeddings. Let h_i denote the patch representations derived from the VQ encoder. We find the nearest codes of each h_i from codebook embeddings $\{v_i | i = 1, \dots, K\}$:

$$z_i = \arg \min_j \| \ell_2(h_i) - \ell_2(v_i) \|_2, \quad (1)$$

where $j \in \{1, \dots, K\}$ and ℓ_2 normalization is employed so that the above distance is equivalent to cosine similarity. Consequently, an EEG sample is tokenized to $z = [z_1, \dots, z_N]$.

Temporal-frequency Prediction. We propose to predict both original signals and the frequency magnitude to capture the temporal and frequency domains of EEG signals. This differs from LaBraM which regresses the Fourier amplitude and phase since we observe that reconstructing the phase contributes minor to neural tokenizer training. We apply the Discrete Fourier Transform (DFT) on an EEG patch $x_{i,j} = [x[1], x[2], \dots, x[P]]$ of channel i and time j , and transform the equation using Euler’s formula as follows

$$\tilde{x}_{i,j}^m = \sum_{n=1}^M x[n] \cos\left(\frac{2\pi}{M}mn\right) - jx[n] \sin\left(\frac{2\pi}{M}mn\right). \quad (2)$$

where $m \in [1, N]$ and j is the imaginary unit. Accordingly, we calculate the frequency magnitude as $f^m = \sqrt{\text{Re}(\tilde{x}_{i,j}^m)^2 + \text{Im}(\tilde{x}_{i,j}^m)^2}$, where Re and Im represent the real and imaginary parts of a complex number. For stable convergence, we adopt z-score normalization to the magnitude within a sample.

After being quantized to the codebook embeddings, we feed the normalized neural embeddings $[\ell_2(z_1), \dots, \ell_2(z_N)]$ into two separate decoders. Let o_i^t and o_i^f stand for the output of a temporal decoder and a frequency decoder, respectively. The optimizing target for the codebook learning is

$$\mathcal{L}_1 = \sum_{x \in \mathcal{D}} \sum_i \underbrace{\|o_i^t - x_i\|_2^2 + \|o_i^f - f_i\|_2^2}_{\text{reconstruction loss}} + \underbrace{\|\text{sg}(\ell_2(h_i)) - \ell_2(v_{z_i})\|_2^2}_{\text{codebook loss}} + \underbrace{\|\ell_2(h_i) - \text{sg}(\ell_2(v_{z_i}))\|_2^2}_{\text{commitment loss}}, \quad (3)$$

where \mathcal{D} represents the whole dataset and sg denotes the stop-gradient operator that is identical during forward computation and has zero partial derivatives.

EEG-text Embedding Space Alignment. Current vision-language models usually utilize pre-trained CLIP-like (Radford et al., 2021) image encoders which are trained by large-scale image-text pairs and thus are embedding-wise well-aligned with text. However, when considering EEG, there are much more challenges to align EEG with text: 1) EEG signals contain complicated cognitive and non-cognitive information, which is hard to be described by human language accurately and thoroughly. For example, an EEG segment can not only contain one person’s emotion and mental states, but also represent the body movement and medical normality. 2) The labeled EEG data available to construct EEG-text pair are very limited. Therefore, we propose to align EEG with text space-wise instead of embedding-wise.

We introduce a domain classifier \mathcal{C} to predict whether the embeddings are from EEG or text. During the codebook learning, we also feed some text embeddings from LLMs to train the domain classifier. A gradient reverse layer (Ganin et al., 2016) is added after the VQ encoder to confuse the domain classifier. Hence, the embeddings from the VQ encoder fall into the same space of text embeddings. Consequently, the training objective for text-aligned neural tokenizer training is defined as

$$\min \mathcal{L}_1 + \lambda \sum_i d_i \log \mathcal{C}(h_i), \quad (4)$$

where d_i is the label of EEG or text domain and $\lambda = \frac{2}{1+e^{-10t/T}} - 1$ is a scaling factor that gradually changes from 0 to 1.

VQ Encoder Architecture. We briefly introduce the architecture of the VQ encoder as it is almost the same as LaBraM. The temporal encoder and spatial encoder are two pivotal parts of the VQ encoder. The temporal encoder contains several blocks of 1-D convolution which aims to extract temporal features in each EEG patch. After that, learnable temporal and spatial embeddings are added according to the standard 10-20 international system to inject both time and channel information. Finally, the spatial encoder composed of vanilla Transformer blocks (Vaswani et al., 2017) learns interaction among patches.

2.2 MULTI-CHANNEL AUTOREGRESSIVE PRE-TRAINING

Before passing EEG data into Large Language Models, we freeze the VQ encoder and first use it to encode input EEG data to EEG tokens that are aligned with the text space. After that, we load a pre-trained Large Language Model and enlarge the text vocabulary with the learned EEG codebook.

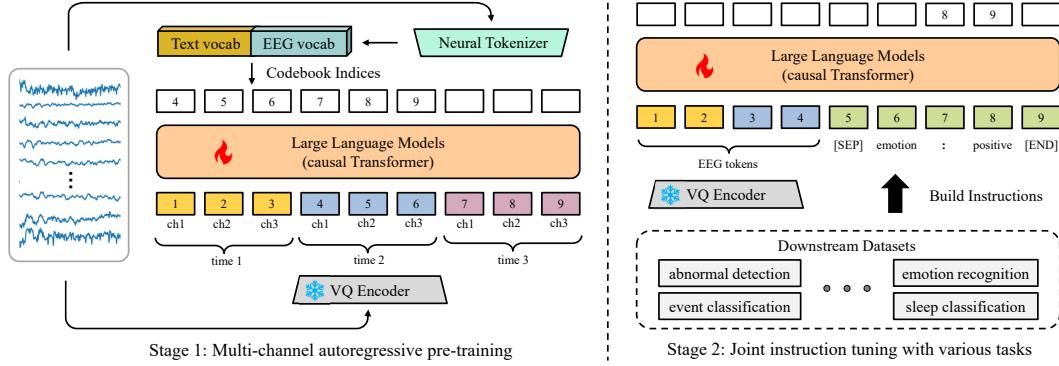


Figure 3: Schematic of NeuroLM training. **Left:** We first pre-train NeuroLM via multi-channel autoregression with EEG tokens output by the frozen VQ encoder. **Right:** The multi-task instruction tuning enables NeuroLM to perform various BCI tasks within a single model.

The EEG tokens are added with reused temporal embeddings from the LLM and new spatial embeddings. As shown in Figure 3, NeuroLM is then trained through multi-channel autoregression, that is, predicting the next EEG tokens based on visible EEG tokens, to endow the model with the capability of learning special patterns of EEG causal relationship. In our experiments, the multi-channel autoregressive pre-training contributes to the performance of multi-task instruction tuning.

Formulation. Consider a sequence of EEG tokens $\mathbf{h} = \{h_{ij}|i = 1, \dots, C, j = 1, \dots, N\}$ where i denotes the channel and j denote the time, and their corresponding indices of the merged text and EEG vocabulary $\mathbf{I} = \{I_{ij}|i = 1, \dots, C, j = 1, \dots, N\}$ derived from the neural tokenizer. Unlike language that can be predicted token by token intuitively, EEG signals are of various configurations, thus it is impracticable to directly predict EEG tokens one by one. We propose a multi-channel autoregressive strategy to adopt the idea of autoregression on EEG. The basic idea is that each token of a specific channel predicts the next token of the same channel, which can be formulated as

$$p(I_{11}, I_{12}, \dots, I_{CN}) = \prod_{n=1}^N p(I_{1n}, I_{2n}, \dots, I_{Cn}|h_{11}, h_{12}, \dots, h_{C(n-1)}). \quad (5)$$

Therefore, the objective for multi-channel autoregressive pre-training is to optimize model parameters by maximizing $p(h_{1n}, h_{2n}, \dots, h_{Cn}|h_{11}, h_{12}, \dots, h_{C(n-1)})$ throughout all EEG data.

For implementation, we define stair-stepping masks where each EEG token is able to observe tokens of all channels from its current and previous time step. Figure 4 illustrates the design of our stair-stepping mask. Dark cells indicate that the elements should take part in attention.

Theory Analysis. We interpret the multi-channel autoregressive pre-training from the view of a variational autoencoder (Kingma & Welling, 2014). Let x denote the original EEG signals, y denote the temporal-frequency target of x , and \hat{x} be the EEG tokens to be predicted. Assume that EEG signals x can be generated by a random process with a latent variable z . We use $q_\phi(z|x_i)$ to denote the VQ encoder encoding EEG signals into discrete neural codes, $p_\psi(y_i|z_i)$ to stand for the temporal and frequency decoder reconstructing temporal-frequency domain from encoded neural codes, and $p_\theta(z|x_i)$ to represent multi-channel autoregressive pre-training. Consider the log-likelihood $p(y|x)$ and its evidence lower bound (ELBO), involving predicting the temporal-frequency domain of the EEG signals from the next time point:

$$\sum_{i=1}^N \log p(y_i|x_i) \geq - \sum_{i=1}^N (\mathbb{E}_{z_i \sim q_\phi(z|x_i)}[-\log p_\psi(y_i|z_i)] + KL(q_\phi(z|x_i), p_\theta(z|x_i))), \quad (6)$$

where the first term is the reconstruction loss and the second term is Kullback-Leibler divergence between q and EEG-text conditional prior. Our training paradigm encompasses two-stage learning

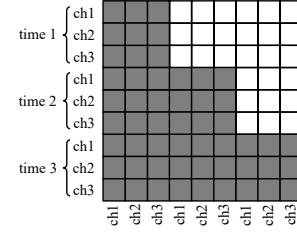


Figure 4: The stair-stepping mask. Each row indicates attention masks for an EEG token.

processes: 1) The neural tokenizer is optimized by minimizing the reconstruction loss. 2) A LLM learns the prior p_θ by minimizing KL loss with q_ϕ and p_ψ fixed. The sequence z_i can be sampled from $q_\phi(z|x_i)$ or one-point distribution $z_i = \arg \max_z q_\phi(z|x_i)$ where we choose the latter for simplicity. Therefore, Equation 6 can be rewritten as

$$-\sum_{i=1}^N (\mathbb{E}_{z_i \sim q_\phi(z|x_i)} [-\log p_\psi(y_i|z_i)] - \log p_\theta(z_i|x_i)), \quad (7)$$

where the latter term is the negative log-likelihood loss for multi-channel autoregressive pre-training.

2.3 MULTI-TASK INSTRUCTION TUNING

In this stage, we aim to leverage the power of LLMs to integrate different downstream datasets as a whole. Instruction tuning is introduced to handle various downstream tasks, as shown in Figure 3. It is worthwhile to mention that in both multi-channel autoregressive pre-training and multi-task instruction tuning stages, we feed the model a few text data at each iteration to preserve the language modeling capability of LLMs. We build instructions for each downstream dataset and the instruction design can be found in Appendix B. A special token [SEP] is used to concatenate EEG and text instructions, indicating the modality switch. Notably, the loss is only calculated on the answer part of the text to make the prediction more stable. Suppose x^p represents the EEG tokens along with the question part of the instruction (prompt), and t^a represents the answer part of the instruction. Let the sequence length of t^a be L , and this procedure can be written as

$$p(t^a|x^p) = \prod_{i=1}^L p(t_i^a|x^p, t_{,<i}^a), \quad (8)$$

where $t_{,<i}^a$ is the answer tokens before the current prediction token t_i^a .

3 EXPERIMENTS

3.1 DOWNSREAM DATASETS

We consider six different EEG datasets with highly varied data sizes to comprehensively evaluate NeuroLM, where the detailed information is listed in Table 1: 1) **TUAB** (Harati et al., 2015) (abnormal detection): This dataset contains EEG records that are classified as clinically normal or abnormal. 2) **TUEV** (Harati et al., 2015) (event type classification): This corpus contains six events involving periodic lateralized epileptiform discharge, generalized periodic epileptiform discharge, spike and/or sharp wave discharges, artifact, and eye movement. 3) **SEED** (Zheng & Lu, 2015) (emotion recognition): There are 3 emotions (positive, negative, and neutral) elicited by videos from 15 subjects. There are 15 trials in each session and each subject underwent 3 sessions. 4) **HMC** (Alvarez-Estevez & Rijsman, 2021) (sleep stage classification): HMC was developed for automatic sleep scoring, involving 5 sleep stages (wake, NREM-1, NREM-2, NREM-3, REM) from 154 subjects. 5) **Workload** (Zyma et al., 2019) (cognitive workload classification): This dataset contains 36 subjects performing serial subtraction. We regard mental workload trials as high workload and the last 60 seconds of the rest EEG as low workload. 6) **TUSL** (von Weltin et al., 2017) (slowing event classification): TUSL aims to differentiate between seizure, slowing, and complex background events.

Table 1: Information of datasets used for downstream evaluation.

Dataset	#Channel	Sampling Rate	Duration	#Sample	Task
TUAB	23	256 Hz	10 seconds	409,455	Binary classification
TUEV	23	256 Hz	5 seconds	112,491	6-class classification
SEED	62	1000 Hz	4 seconds	38,475	3-class classification
HMC	4	256 Hz	30 seconds	137,243	5-class classification
Workload	19	500 Hz	4 seconds	2,088	Binary classification
TUSL	23	256 Hz	10 seconds	245	3-class classification

For the data division, we split each dataset into training, validation, and test sets: 1) **TUAB** and **TUEV**: Since the training and test division is provided by the original datasets, we further divide the training patients into training and validation groups by 80% and 20% randomly. 2) **SEED**: We split total 15 trials into training, validation, and test trials by 9:3:3 according to the chronological order, and merge all sessions into the final training, validation, and test set. 3) **HMC**: subjects from number 1 to 100 form the training set while subjects from number 101 to 126 and number 126 to 154 are validation and test sets, respectively. 4) **Workload**: The training, validation, and test sets are derived by subjects from number 0 to 25, number 26 to 30, and number 31 to 35, respectively. 5) **TUSL**: The training, validation, and test sets are splitted by 60%:20%:20%.

3.2 EXPERIMENTAL SETUP

Model Configurations. NeuroLM is compatible with any causal LLM as its base language model. For simplicity and saving computing resources, we adopt GPT-2 (Radford et al., 2019) as our base language model. Accordingly, NeuroLM has three variants, NeuroLM-B, NeuroLM-L, and NeuroLM-XL, which have 254M, 500M, and 1696M parameters (including the parameters of the VQ encoder), respectively. Unless otherwise noted, NeuroLM refers to NeuroLM-B. The patch size P is set to 200 (1 second), consistent with that of LaBraM. To maintain compatibility with GPT-2, the maximum sequence length (number of patches) is set to 1024. For input samples with sequence lengths shorter than 1024, we pad zeros to ensure the length is equal to 1024 at neural tokenizer training and multi-channel autoregressive pre-training stages. The attention values for these zero paddings will be masked.

Data Preprocessing. To eliminate environmental and physiological artifacts from EEG signals, we employ several necessary preprocessing methods. First, we apply a bandpass filter with cutoff frequencies of 0.1 Hz and 75 Hz to remove low-frequency noise. To avoid power-line interference, we use a notch filter at 50 Hz or 60 Hz, depending on the geographic region of data collection. Additionally, all signals are resampled to 200 Hz to reduce computational complexity. Given that EEG signal values typically range between $-100 \mu\text{V}$ to $100 \mu\text{V}$, all values are divided by 100 for normalization.

Training & Environment Settings. To facilitate the training of NeuroLM, a huge volume of data is required. About 25,000 hours of EEG data from multiple public EEG datasets are collected after cleaning and filtering, which are listed in Appendix C. All experiments are conducted on eight NVIDIA A100-80G GPUs with Python 3.11.8 and PyTorch 2.2.2 + CUDA 12.1. For instruction tuning, the results are obtained using the final model after training. Notably, we choose the largest logits as the prediction at evaluation and test instead of beam search which is widely used in current LLMs to obtain stable results. For other scenarios, the baselines are in a single-task manner and trained on individual datasets. Their best models are selected based on the best performance on the validation set, and then evaluated on the test set. The average and standard deviation values are reported using three random seeds to ensure comparable results. Baselines and other detailed hyperparameter settings are provided in Appendix D.

3.3 EXPERIMENTAL RESULTS

We present all results in Table 2, 3, and 4. Underlined values represent the best results for single-task methods, while bold values indicate the best results for NeuroLM. Notably, it's important to note that direct comparisons between NeuroLM and the baseline single-task methods are not entirely fair, as the baselines are trained and tested on individual datasets. Although NeuroLM is still a few steps away from the state-of-the-art LaBraM, it achieves performance comparable to most other single-task baselines. Specifically, NeuroLM performs similarly to BIOT and surpasses all other baselines except LaBraM on TUEV. Additionally, NeuroLM ranks second on TUSL. For NeuroLM-L and NeuroLM-XL, the performance is further enhanced on most downstream datasets with larger model capacity. However, the imbalance in data size among different downstream datasets poses a challenge for NeuroLM, as it reaches optimal performance at different training times for different datasets. Additionally, we find that models with more parameters are more prone to overfitting, which might account for the performance degradation observed on HMC since sleep patterns are of low complexity and smaller models might be sufficient to capture the relevant features in EEG

Table 2: Results on TUAB and TUEV.

Methods	Multi-task	TUAB			TUEV		
		Balanced Acc.	AUC-PR	AUROC	Balanced Acc.	Cohen's Kappa	Weighted F1
SPaRCNet	✗	0.7896±0.0018	0.8414±0.0018	0.8676±0.0012	0.4161±0.0262	0.4233±0.0181	0.7024±0.0104
ContraWR	✗	0.7746±0.0041	0.8421±0.0104	0.8456±0.0074	0.4384±0.0349	0.3912±0.0237	0.6893±0.0136
CNN-Transformer	✗	0.7777±0.0022	0.8433±0.0039	0.8461±0.0013	0.4087±0.0161	0.3815±0.0134	0.6854±0.0293
FFCL	✗	0.7848±0.0038	0.8448±0.0065	0.8569±0.0051	0.3979±0.0104	0.3732±0.0188	0.6783±0.0120
ST-Transformer	✗	0.7966±0.0023	0.8521±0.0026	0.8707±0.0019	0.3984±0.0228	0.3765±0.0306	0.6823±0.0190
BIOT	✗	0.7959±0.0057	0.8792±0.0023	0.8815±0.0043	0.5281±0.0225	0.5273±0.0249	0.7492±0.0082
LaBraM	✗	<u>0.8140</u> ±0.0019	<u>0.8965</u> ±0.0016	<u>0.9022</u> ±0.0009	<u>0.6409</u> ±0.0065	<u>0.6637</u> ±0.0093	<u>0.8312</u> ±0.0052
NeuroLM-B	✓	0.7826±0.0065	0.6975±0.0081	0.7816±0.0079	0.4560±0.0048	0.4285±0.0048	0.7153±0.0028
NeuroLM-L	✓	0.7876±0.0034	0.7099±0.0034	0.7876±0.0034	0.4132±0.1235	0.4414±0.0996	0.7387 ±0.0400
NeuroLM-XL	✓	0.7969 ±0.0091	0.7219 ±0.0082	0.7884 ±0.0194	0.4679 ±0.0356	0.4570 ±0.0498	0.7359±0.0219

Table 3: Results on SEED and HMC.

Methods	Multi-task	SEED			HMC		
		Balanced Acc.	Cohen's Kappa	Weighted F1	Balanced Acc.	Cohen's Kappa	Weighted F1
SPaRCNet	✗	0.5596±0.0244	0.3464±0.0372	0.5585±0.0297	0.4756±0.1109	0.3147±0.1315	0.4108±0.1310
ContraWR	✗	0.6106±0.0078	0.4220±0.0129	0.6137±0.0085	0.4242±0.0541	0.2340±0.0554	0.2987±0.0288
CNN-Transformer	✗	0.6161±0.0384	0.4262±0.0601	0.6150±0.0463	0.6573±0.0141	0.5961±0.0105	0.6896±0.0065
FFCL	✗	0.5808±0.0322	0.3732±0.0462	0.5743±0.0402	0.4427±0.0702	0.2542±0.0654	0.2902±0.0485
ST-Transformer	✗	0.5479±0.0091	0.3261±0.0169	0.5505±0.0091	0.2559±0.0141	0.0503±0.0183	0.1428±0.0122
BIOT	✗	0.7097±0.0024	0.5682±0.0051	0.7134±0.0027	0.6862±0.0041	0.6295±0.0113	0.7091±0.0147
LaBraM	✗	<u>0.7318</u> ±0.0019	<u>0.5994</u> ±0.0031	<u>0.7354</u> ±0.0021	<u>0.7286</u> ±0.0101	<u>0.6812</u> ±0.0073	<u>0.7554</u> ±0.0024
NeuroLM-B	✓	0.5554±0.0075	0.3393±0.0117	0.5599±0.0068	0.6737 ±0.0050	0.6188 ±0.0057	0.7126 ±0.0034
NeuroLM-L	✓	0.6006±0.0047	0.4067±0.0063	0.6048±0.0050	0.6658±0.0050	0.5929±0.0715	0.6896±0.0504
NeuroLM-XL	✓	0.6034 ±0.0010	0.4082 ±0.0036	0.6063 ±0.0030	0.5761±0.1084	0.4795±0.1466	0.5883±0.1286

Table 4: Results on Workload and TUSL.

Methods	Multi-task	Workload			TUSL		
		Balanced Acc.	AUC-PR	AUROC	Balanced Acc.	Cohen's Kappa	Weighted F1
SPaRCNet	✗	0.5977±0.0071	0.6638±0.0314	0.6717±0.0172	0.4185±0.0452	0.1399±0.0799	0.3500±0.0968
ContraWR	✗	0.6966±0.0332	0.7668±0.0408	0.7685±0.0317	0.5857±0.0662	0.3567±0.0968	0.5458±0.0798
CNN-Transformer	✗	0.5793±0.0230	0.5306±0.0459	0.5663±0.0349	0.3575±0.0151	0.0306±0.0179	0.2235±0.0251
FFCL	✗	<u>0.7069</u> ±0.0197	<u>0.7823</u> ±0.0099	<u>0.7857</u> ±0.0234	0.3819±0.0688	0.0628±0.0888	0.2120±0.0786
ST-Transformer	✗	0.6103±0.0056	0.5716±0.0071	0.6375±0.0078	0.4000±0.0329	0.0860±0.0449	0.3793±0.0459
BIOT	✗	0.6655±0.0665	0.7189±0.0722	0.7342±0.0536	0.5758±0.0303	0.2012±0.0212	0.2394±0.0040
LaBraM	✗	0.6609±0.0204	0.7174±0.0234	0.7272±0.0165	<u>0.7625</u> ±0.0131	<u>0.6407</u> ±0.0304	<u>0.7614</u> ±0.0210
NeuroLM-B	✓	0.6172±0.0113	0.5824±0.0080	0.6253 ±0.0160	0.6734±0.0436	0.5107±0.0617	0.6743±0.0394
NeuroLM-L	✓	0.6311±0.0250	0.5869±0.0155	0.6247±0.0339	0.5314±0.0530	0.2961±0.0810	0.5243±0.0680
NeuroLM-XL	✓	0.6345 ±0.0442	0.5889 ±0.0423	0.6130±0.0764	0.6845 ±0.0304	0.5194 ±0.0461	0.6839 ±0.0297

signals. On TUSL, the model performance appears to be not very stable due to the extremely limited data samples.

3.4 ABLATION ON ROBUSTNESS

Our instruction design for some datasets (TUEV, HMC, and TUSL) follows multiple-choice questions. To validate the robustness of NeuroLM, we enumerate the orders of options and randomly select one from all possible combinations during data fetching of the multi-task instruction tuning stage. Figure 5 illustrates the results on whether shuffling the options. We can conclude that on TUEV and HMC, NeuroLM with shuffle obtains comparable performance compared to those without shuffle. Nevertheless, it seems that the shuffle operation significantly degrades the performance on TUSL. We attribute this phenomenon to the lack of data for TUSL because TUSL has much fewer number of data samples compared to the other two datasets. It is expected that NeuroLM will achieve similar results if given more data. In general, NeuroLM has good robustness against arbitrary order of options, which indicates that NeuroLM does understand the linguistic meaning of the questions when predicting.

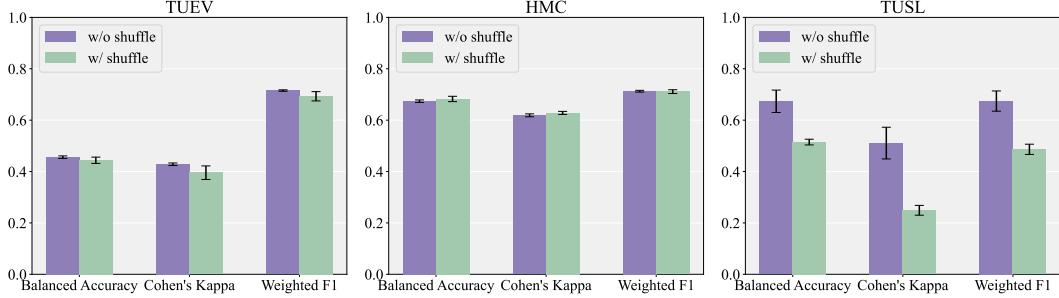


Figure 5: Ablation study on whether shuffling the options of instructions.

3.5 ABLATION ON INSTRUCTION DATA SIZE

We utilize TUAB, TUEV, and HMC datasets to scale the instruction data size and validate the performance of NeuroLM and other baseline methods, as these three datasets have a relatively large number of samples. The results, illustrated in Figure 6, show that NeuroLM demonstrates consistent performance under all conditions. For TUAB, NeuroLM, LaBraM, and CNN-Transformer exhibit stable performance. For TUEV and HMC, only NeuroLM and LaBraM are relatively unaffected by changes in data size. These findings indicate that NeuroLM is robust and maintains high performance even with varying instruction data sizes, highlighting its effectiveness in multi-task learning scenarios.

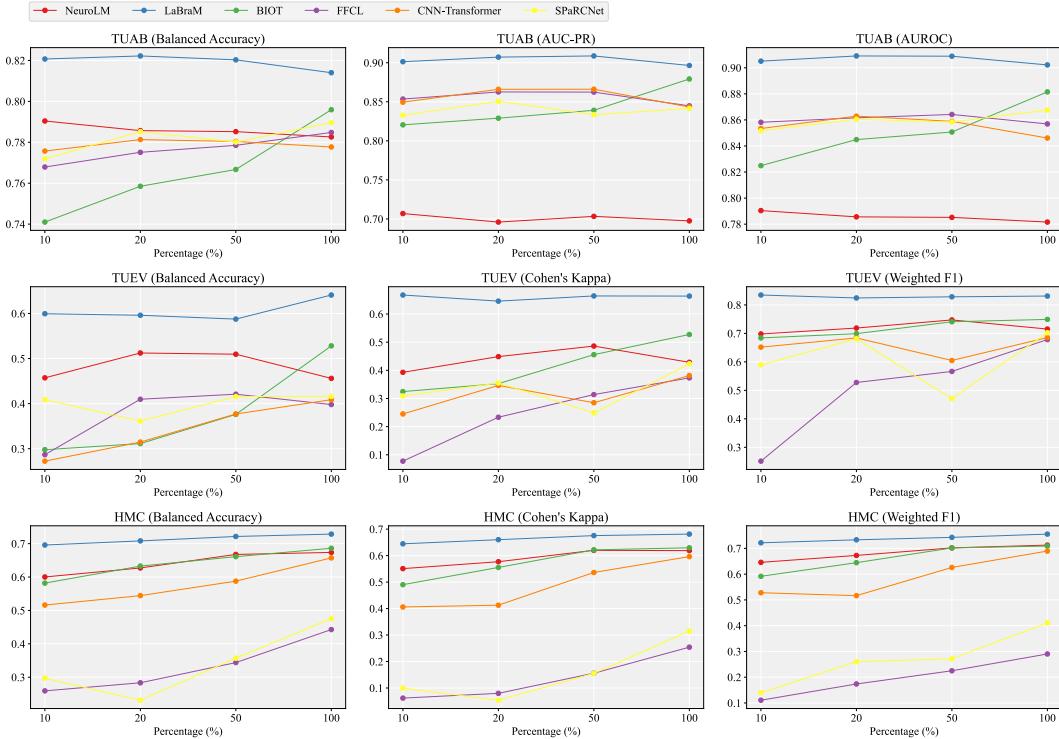


Figure 6: Comparison of different methods under different proportions of instruction data.

3.6 VISUALIZATION CURVES OF MULTI-CHANNEL AUTOREGRESSION

We visualize the pre-training loss, accuracy, and validation perplexity of NeuroLM in Figure 7. We observe that the loss stably converges while the validation perplexity decreases with training, which means NeuroLM can generalize well to unseen EEG data. Intuitively, a larger model with more parameters obtains smaller loss and perplexity. Additionally, NeuroLM-L achieves similar

validation perplexity with NeuroLM-XL, indicating that current pre-training data size still cannot satisfy the training with billion-level parameters.

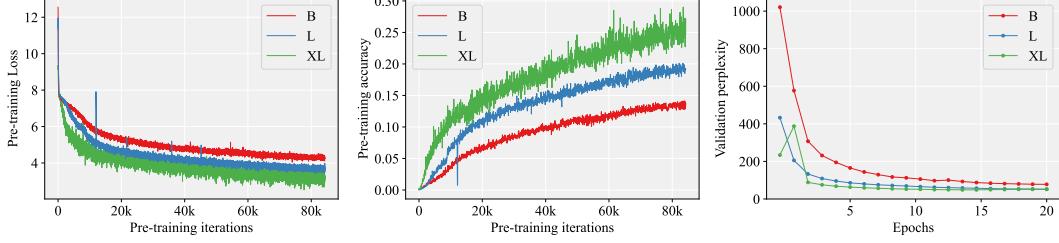


Figure 7: The training and validation visualization of multi-channel autoregressive pre-training.

3.7 ABLATION ON MULTI-CHANNEL AUTOREGRESSIVE PRE-TRAINING

The proposed multi-channel autoregressive pre-training aims at mimicing current causal LLMs by predicting the next EEG tokens for each channel. It is expected to benefit downstream tasks through learning causal representations. We perform an ablation study to assess the impact of the proposed multi-channel autoregressive pre-training on NeuroLM. The results, shown in Figure 8, reveal a significant performance improvement when NeuroLM is pre-trained with this approach, underscoring the effectiveness of multi-channel autoregressive pre-training.

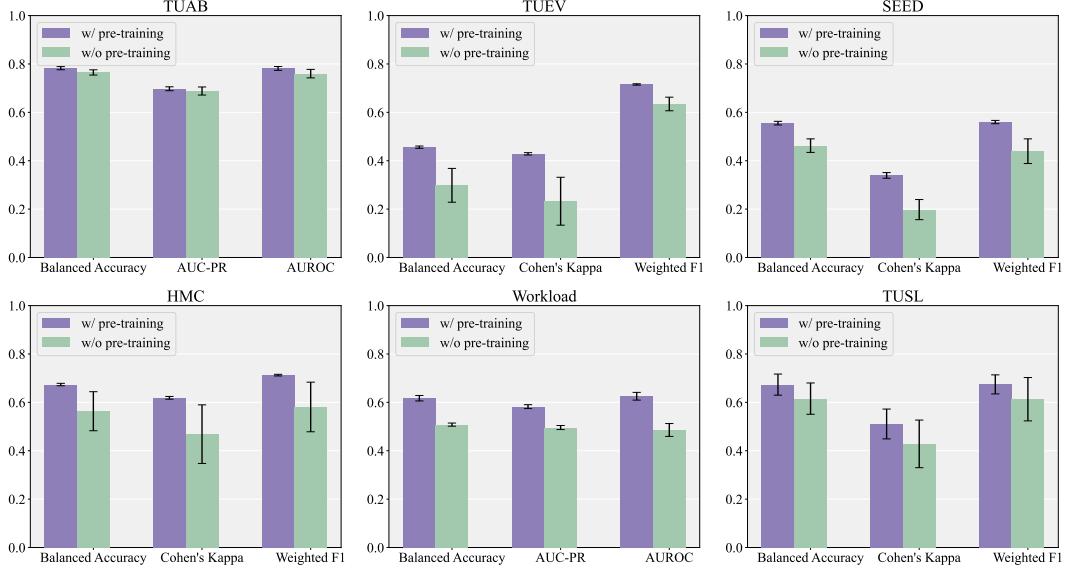


Figure 8: Ablation study on multi-channel autoregressive pre-training.

4 CONCLUSION

In this paper, we introduce NeuroLM, the first universal multi-task foundation model for EEG signal processing. By integrating EEG signals into a Large Language Model framework, NeuroLM leverages advanced text-aligned neural tokenizer embeddings, large-scale multi-channel autoregressive pre-training, and joint multi-task tuning to address the inherent challenges of EEG-based BCI and healthcare tasks. Our extensive experiments across six diverse EEG datasets demonstrate the model’s superior performance in multi-task learning and inference. Overall, NeuroLM represents a significant step forward in the field of brain-computer interfaces and healthcare domains, showcasing the great potential of LLMs to revolutionize EEG signal processing and multi-task learning. We believe that NeuroLM will pave the way for more sophisticated and versatile EEG applications, ultimately enhancing the interaction between humans and machines.

REFERENCES

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- Turkey N Alotaiby, Saleh A Alshebeili, Tariq Alshawi, Ishtiaq Ahmad, and Fathi E Abd El-Samie. EEG seizure detection and prediction algorithms: a survey. *EURASIP Journal on Advances in Signal Processing*, 2014:1–21, 2014.
- Diego Alvarez-Estevez and Roselyne M Rijsman. Inter-database validation of a deep learning approach for automatic sleep scoring. *PloS one*, 16(8):e0256111, 2021.
- Benjamin Blankertz, Guido Dornhege, Matthias Krauledat, Klaus-Robert Müller, and Gabriel Curio. The non-invasive berlin brain–computer interface: fast acquisition of effective performance in untrained subjects. *NeuroImage*, 37(2):539–550, 2007.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*, 2024.
- Paolo Detti, Giampaolo Vatti, and Garazi Zabalo Manrique de Lara. Eeg synchronization analysis for seizure prediction: A study on data of noninvasive recordings. *Processes*, 8(7):846, 2020.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016. URL <http://jmlr.org/papers/v17/15-239.html>.
- Zhongke Gao, Xinmin Wang, Yuxuan Yang, Chaoxu Mu, Qing Cai, Weidong Dang, and Siyang Zuo. EEG-based spatio-temporal convolutional neural network for driver fatigue evaluation. *IEEE Transactions on Neural Networks and Learning Systems*, 30(9):2755–2763, 2019.
- A. Harati, M. Golmohammadi, S. Lopez, I. Obeid, and J. Picone. Improved EEG event classification using differential energy. In *2015 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pp. 1–4, 2015. doi: 10.1109/SPMB.2015.7405421.
- Robert Jenke, Angelika Peer, and Martin Buss. Feature Extraction and Selection for Emotion Recognition from EEG. *IEEE Transactions on Affective Computing*, 5(3):327–339, 2014. doi: 10.1109/TAFFC.2014.2339834.
- Wei-Bang Jiang, Li-Ming Zhao, and Bao-Liang Lu. Large brain model for learning generic representations with tremendous EEG data in BCI. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=QzTpTRVtrP>.
- Jin Jing, Wendong Ge, Shenda Hong, Marta Bento Fernandes, Zhen Lin, Chaoqi Yang, Sungtae An, Aaron F Struck, Aline Herlopian, Ioannis Karakis, et al. Development of expert-level classification of seizures and rhythmic and periodic patterns during eeg interpretation. *Neurology*, 100(17):e1750–e1762, 2023.
- Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

- Louis Korczowski, Martine Cederhout, Anton Andreev, Grégoire Cattan, Pedro Luiz Coelho Rodrigues, Violette Gautheret, and Marco Congedo. Brain Invaders calibration-less P300-based BCI with modulation of flash duration Dataset (bi2015a). Research report, GIPSA-lab, July 2019. URL <https://hal.science/hal-02172347>.
- Demetres Kostas, Stephane Aroca-Ouellette, and Frank Rudzicz. Bendr: using transformers and a contrastive self-supervised learning task to learn from massive amounts of eeg data. *Frontiers in Human Neuroscience*, 15:653659, 2021.
- Hongli Li, Man Ding, Ronghua Zhang, and Chunbo Xiu. Motor imagery EEG classification algorithm based on CNN-LSTM feature fusion network. *Biomedical signal processing and control*, 72:103342, 2022.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=w0H2xGH1kw>.
- Wei Liu, Jie-Lin Qiu, Wei-Long Zheng, and Bao-Liang Lu. Comparing recognition performance and robustness of multimodal deep learning models for multimodal emotion recognition. *IEEE Transactions on Cognitive and Developmental Systems*, 2021.
- Wei Liu, Wei-Long Zheng, Ziyi Li, Si-Yuan Wu, Lu Gan, and Bao-Liang Lu. Identifying similarities and differences in emotion recognition with eeg and eye movements among chinese, german, and french people. *Journal of Neural Engineering*, 19(2):026012, 2022.
- Matthew D Luciw, Ewa Jarocka, and Benoni B Edin. Multi-channel EEG recordings during 3,936 grasp and lift trials with varying weight and friction. *Scientific Data*, 1(1):1–11, 2014.
- Perrin Margaux, Maby Emmanuel, Daligault Sébastien, Bertrand Olivier, and Mattout Jérémie. Objective and subjective evaluation of online error correction during p300-based spelling. *Advances in Human-Computer Interaction*, 2012:4–4, 2012.
- Iyad Obeid and Joseph Picone. The temple university hospital eeg data corpus. *Frontiers in neuroscience*, 10:195498, 2016.
- Wei Yan Peh, Yuanyuan Yao, and Justin Dauwels. Transformer Convolutional Neural Networks for Automated Artifact Detection in Scalp EEG. In *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 3599–3602, 2022. doi: 10.1109/EMBC48229.2022.9871916.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.
- Arman Savran, Koray Ciftci, Guillame Chanel, Javier Cruz_Mota, Luong Hong Viet, Bülent Sankur, Lale Akarun, Alice Caplier, and Michele Rombaut. Emotion detection in the loop from brain signals and facial images. In *eINTERFACE'06-SIMILAR NoE Summer Workshop on Multimodal Interfaces*, 2006.
- Gerwin Schalk, Dennis J McFarland, Thilo Hinterberger, Niels Birbaumer, and Jonathan R Wolpaw. BCI2000: a general-purpose brain-computer interface (BCI) system. *IEEE Transactions on Biomedical Engineering*, 51(6):1034–1043, 2004.
- Yonghao Song, Xueyu Jia, Lie Yang, and Longhan Xie. Transformer-based spatial-temporal feature learning for EEG decoding. *arXiv preprint arXiv:2106.11170*, 2021.
- Akara Supratak, Hao Dong, Chao Wu, and Yike Guo. DeepSleepNet: A Model for Automatic Sleep Stage Scoring Based on Raw Single-Channel EEG. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(11):1998–2008, 2017. doi: 10.1109/TNSRE.2017.2721116.

- Yousef Rezaei Tabar and Ugur Halici. A novel deep learning approach for classification of EEG motor imagery signals. *Journal of Neural Engineering*, 14(1):016003, 2016.
- Mastaneh Torkamani-Azar, Sumeyra Demir Kanik, Serap Aydin, and Mujdat Cetin. Prediction of reaction time and vigilance variability from spatio-spectral features of resting-state EEG in a long sustained attention task. *IEEE Journal of Biomedical and Health Informatics*, 24(9):2550–2558, 2020.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.
- Logan Trujillo. Raw EEG Data. 2020. doi: 10.18738/T8/SS2NHB. URL <https://doi.org/10.18738/T8/SS2NHB>.
- Logan T Trujillo, Candice T Stanfield, and Ruben D Vela. The effect of electroencephalogram (EEG) reference choice on information-theoretic measures of the complexity and integration of EEG signals. *Frontiers in Neuroscience*, 11:425, 2017.
- Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in Neural Information Processing Systems*, 30, 2017.
- Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(11), 2008.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fb053c1c4a845aa-Paper.pdf.
- Eva von Weltin, Tameem Ahsan, Vinit Shah, Dawer Jamshed, Meysam Golmohammadi, Iyad Obeid, and Joseph Picone. Electroencephalographic slowing: A primary source of error in automatic seizure detection. In *2017 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pp. 1–5. IEEE, 2017.
- Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, et al. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*, 2023.
- Chaoqi Yang, M Brandon Westover, and Jimeng Sun. BIOT: Biosignal transformer for cross-data learning in the wild. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a. URL <https://openreview.net/forum?id=c2LZyTyddi>.
- Chaoqi Yang, Cao Xiao, M Brandon Westover, Jimeng Sun, et al. Self-supervised electroencephalogram representation learning for automatic sleep staging: model development and evaluation study. *JMIR AI*, 2(1):e46769, 2023b.
- Ke Yi, Yansen Wang, Kan Ren, and Dongsheng Li. Learning topology-agnostic EEG representations with geometry-aware modeling. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=hiOUySN0ub>.
- Daoze Zhang, Zhizhang Yuan, Yang Yang, Junru Chen, Jingjing Wang, and Yafeng Li. Brant: Foundation model for intracranial neural signal. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=DDkl9vaJyE>.
- W. Zheng, W. Liu, Y. Lu, B. Lu, and A. Cichocki. Emotionmeter: A multimodal framework for recognizing human emotions. *IEEE Transactions on Cybernetics*, pp. 1–13, 2018. ISSN 2168-2267. doi: 10.1109/TCYB.2018.2797176.
- Wei-Long Zheng and Bao-Liang Lu. Investigating critical frequency bands and channels for EEG-based emotion recognition with deep neural networks. *IEEE Transactions on Autonomous Mental Development*, 7(3):162–175, 2015. doi: 10.1109/TAMD.2015.2431497.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

Igor Zyma, Sergii Tukaev, Ivan Seleznov, Ken Kiyono, Anton Popov, Mariia Chernykh, and Oleksii Shpenkov. Electroencephalograms during mental arithmetic task performance. *Data*, 4(1):14, 2019.

A RELATED WORK

Large-scale Pre-training for Neural Signals. With the success of self-supervised learning in computer vision and natural language processing, several studies have emerged to learn effective EEG representations. Kostas et al. (2021) first propose BENDER, which adapts contrastive learning to derive compressed representations from massive EEG datasets. MMM (Yi et al., 2023) introduces a pre-training framework with multi-dimensional position encoding, multi-level channel hierarchy, and a multi-stage pre-training strategy to learn topology-agnostic representations. BIOT (Yang et al., 2023a) tokenizes diverse biosignals into unified segments, enabling cross-data learning despite mismatched channels, variable lengths, and missing values. Brant (Zhang et al., 2023) pre-trains on a large corpus of private intracranial EEG data, capturing long-term dependencies, spatial correlations, and both time and frequency domains. Following BIOT, LaBrA (Jiang et al., 2024) further leverages large-scale 2,500 hours public EEG data, and innovatively introduces a neural tokenizer that encodes continuous EEG signals into discrete codes for masked EEG modeling, thus obtaining a considerable improvement. Unfortunately, all these methods require fine-tuning for specific downstream tasks and cannot perform multi-task learning and inference.

Multimodal Large Language Models. Recent years have seen remarkable achievements of LLMs. In light of the complementarity between language and other modalities, Multimodal Large Language Models have been a rising hotspot. The release of GPT-4 (Achiam et al., 2023) shows the extraordinary multimodal understanding and generation abilities, thus leading to a research frenzy over MLLMs. LLaVA (Liu et al., 2023) connects a vision encoder and an LLM, introducing the idea of visual instruction tuning for general-purpose visual and language understanding. Similarly, Zhu et al. (2023) propose MiniGPT-4, which aligns a frozen visual encoder with a frozen advanced LLM, presenting numerous advanced multi-modal abilities. Different from the above MLLMs, CogVLM (Wang et al., 2023) bridges the gap between the frozen pre-trained LLM and visual encoder by a trainable visual expert in the attention and FFN layers. Chen et al. (2024) present InternVL-1.5, closing the capability gap between open-source and proprietary commercial MLLMs by utilizing a strong vision encoder, dynamic high-resolution, and high-quality bilingual dataset.

B INSTRUCTION DESIGN

Table 5: Information of instruction design for downstream datasets.

Dataset	Instruction Description
TUAB	[SEP] Question: Is this EEG segment abnormal? Answer: {Yes, No} [END]
TUEV	[SEP] Question: Which event type does this EEG segment belong to? Options: (A) spike and slow wave. (B) generalized periodic epileptiform discharge. (C) periodic lateralized epileptiform discharge. (D) eye movement. (E) artifact. (F) background. Answer: {(A), (B), (C), (D), (E), (F)} [END]
SEED	[SEP] Question: Which emotion type does this EEG segment belong to? Answer: {Positive, Neutral, Negative} [END]
HMC	[SEP] Question: Which sleep type does this EEG segment belong to? Options: (A) Wake. (B) NREM-1. (C) NREM-2. (D) NREM-3. (E) REM. Answer: {(A), (B), (C), (D), (E)} [END]
Workload	[SEP] Question: Is this EEG segment of high workload? Answer: {Yes, No} [END]
TUSL	[SEP] Question: Which type does this EEG segment belong to? Options: (A) background. (B) seizure. (C) slowing. Answer: {(A), (B), (C)} [END]

C PRE-TRAINING DATASET DESCRIPTION

We utilize multiple EEG datasets with various configurations. The detail information of all the datasets are listed in 6. The total time after data cleaning is close to 25,000 hours.

Table 6: Information of datasets used for pre-training.

Dataset	#Channel	Rate (Hz)	Time (h)	Description
TUEG (Obeid & Picone, 2016)	17-23	250-1024	~24,000	A corpus of 26,846 clinical EEG recordings collected at Temple University Hospital.
SEED Series (Zheng et al., 2018; Liu et al., 2021; 2022)	62	1000	170.54	These datasets including SEED-IV (15 subjects), SEED-V (20 subjects), SEED-GER (8 subjects), and SEED-FRA (8 subjects) in response to emotional videos.
BCI Competition IV-1 (Blankertz et al., 2007)	59	1000	8.21	A motor imagery dataset containing 2 classes of left hand, right hand, foot (+ idle state) for 7 subjects.
Emobrain (Savran et al., 2006)	64	1024	4.94	A multimodal emotion dataset including 16 subjects. The emotions were elicited through a selected subset of the IAPS dataset.
Grasp and Lift (Luciw et al., 2014)	32	500	11.72	A dataset containing 12 subjects performing grasp-and-lift (GAL) trials.
Inria BCI (Margaux et al., 2012)	56	600	29.98	A P300-based spelling dataset including 26 subjects.
Motor Movement/Imagery (Schalk et al., 2004)	64	160	47.3	A motor imagery dataset consisting of 109 volunteers performing 2 baseline tasks, motor movement, and motor imagery.
Raw EEG Data (Trujillo, 2020)	64	256	34.35	EEG was recorded during reported Information-Integration categorization and reported multidimensional Rule-Based categorization tasks.
Resting State (Trujillo et al., 2017)	64	256	3.04	A dataset comprising 22 subjects for a resting task of 8 mins with 4 mins of eyes closed and 4 mins of eyes open.
Siena Scalp EEG Database (Detti et al., 2020)	31	512	30.47	A database consisting of 14 patients.
SPIS Resting State (Torkamani-Azar et al., 2020)	64	2048	0.83	A dataset including 10 subjects, 2.5 minutes recording in eyes-closed and eyes-open prior to a 105-minute session of Sustained Attention to Response Task with fixed-sequence and varying ISIs.
Target Versus Non-Target (Korczowski et al., 2019)	32	512	16	A dataset including 50 subjects playing Brain Invaders, a visual P300 Brain-Computer Interface using oddball paradigm with adaptive Riemannian Geometry (no-calibration).
Self-collected EEG corpus	62	1000	342.23	A mixed self-collected EEG datasets of more than 140 subjects under various conditions.

D DETAILED EXPERIMENTAL SETTINGS

D.1 HYPERPARAMETER SETTINGS

Table 7: Hyperparameters for neural tokenizer.

Hyperparameters	Values
Temporal Encoder	Input channels {1,16,16}
	Output channels {16,16,16}
	Kernel size {15,3,3}
	Stride {8,1,1}
	Padding {7,1,1}
Transformer encoder layers	12
Transformer decoder layers	3
Hidden size	768
MLP size	3072
Attention head number	12
Codebook size	8192×128
Batch size	512
Peak learning rate	5e-5
Minimal learning rate	1e-5
Learning rate scheduler	Cosine
Optimizer	AdamW
Adam β	(0.9,0.999)
Weight decay	1e-4
Total epochs	50
Warmup epochs	5
Data overlap	None
Gradient clipping	None

Table 8: Hyperparameters for autoregressive pre-training.

Hyperparameters	NeuroLM-B	NeuroLM-L	NeuroLM-XL
Model size	254M	500M	1696M
Transformer encoder layers	12	24	48
Hidden size	768	1024	1600
MLP size	3072	4096	6400
Attention head number	12	16	25
EEG batch size	480 (B), 512 (L, XL)		
Text batch size	32 (B), 64 (L, XL)		
Peak learning rate	6e-4		
Minimal learning rate	6e-5		
Learning rate scheduler	Cosine		
Optimizer	AdamW		
Adam β	(0.9,0.95)		
Weight decay	0.1		
Total epochs	20		
Warmup epochs	2		
Data overlap	None		
Gradient clipping	1		

Table 9: Hyperparameters for instruction tuning.

Hyperparameters	Values
Instruction batch size	512
Text batch size	128
Peak learning rate	5e-4 (B), 5e-5 (L), 2e-5 (XL)
Minimal learning rate	5e-5 (B), 5e-6 (L), 2e-6 (XL)
Learning rate scheduler	Cosine
Optimizer	AdamW
Adam β	(0.9,0.95)
Weight decay	0.1
Total epochs	5 (B, L), 3 (XL)
Warmup ratio	0.1
Gradient clipping	1

D.2 METRICS

Considering the class imbalance of most downstream EEG datasets, we use the following metrics for comparison:

- **Balanced Accuracy:** The average of recall (sensitivity) obtained on each class. It is particularly useful for evaluating classification performance on imbalanced datasets. This metric is particularly useful when evaluating models on imbalanced datasets.
- **AUC-PR:** A performance measurement for binary classification problems. It is the area under the curve plotted with precision (y-axis) against recall (x-axis) for different threshold values.
- **AUROC:** It is the area under the curve plotted with the true positive rate (sensitivity) on the y-axis and the false positive rate (1 - specificity) on the x-axis for different threshold values. AUROC provides an aggregate measure of performance across all possible classification thresholds, indicating the ability of the model to distinguish between classes.
- **Cohen’s Kappa:** A measure of agreement between categorical variables X and Y , calculated from the observed and expected frequencies on the diagonal of a square contingency table. It is used for multi-class classification.
- **Weighted F1:** The weighted F1 score is the harmonic mean of precision and recall, taking into account the support (the number of true instances) of each class. The weighted F1 score accounts for class imbalance by giving more importance to classes with a higher number of instances.

AUROC and Cohen’s Kappa are used as the monitor score for binary classification and multi-class classification, respectively.

D.3 BASELINES

We mainly consider BIOT (Yang et al., 2023a) and the state-of-the-art EEG foundation model LaBraM (Jiang et al., 2024) as our baseline method, where BIOT is a generic biosignal learning model pre-trained on multiple datasets in a supervised way, and LaBraM is pre-trained on 2,500 hours data through masked EEG modeling and has learned generic representations for various EEG signals. Five other supervised methods including SPaRCNet (Jing et al., 2023), ContraWR (Yang et al., 2023b), CNN-Transformer (Peh et al., 2022), FFCL (Li et al., 2022), and ST-Transformer (Song et al., 2021) are also utilized as our baselines. As there are no multi-task methods available in EEG signal processing yet, these baselines are solely fine-tuned on each downstream dataset and cannot perform multiple tasks. We use the default settings for these baselines in the BIOT paper. The batch size is 512 for TUAB, TUEV, SEED, and HMC. As the data size of Workload and TUSL is particularly small, the batch size of these two datasets is set to 32 and 16, respectively.

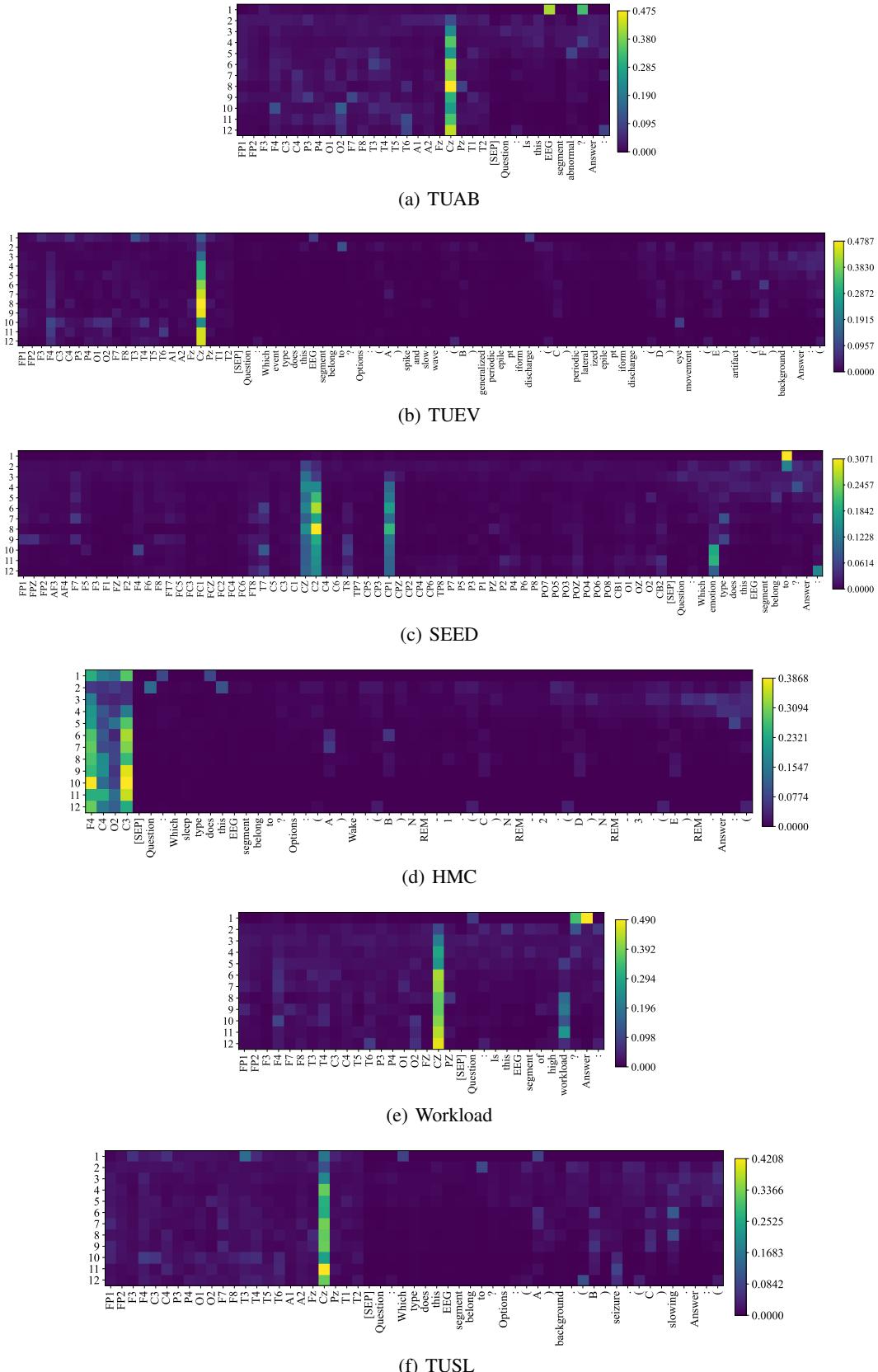


Figure 9: The attention value on other datasets. The vertical axis denotes the Transformer layers.

E ATTENTION VISUALIZATION

To explore the mechanism of NeuroLM, we visualize the attention scores of the answer parts in the instructions for all 12 Transformer layers, as drawn in Figure 9. Firstly, we observed several commonalities across datasets: For the text part, the attention tends to concentrate more in shallow layers whereas for the EEG part, attention gains more in deeper layers. This pattern suggests that NeuroLM primarily processes text questions in the shallow layers and focuses on EEG tokens in the deeper layers to generate answers. Interestingly, in the case of multiple-choice questions, NeuroLM pays close attention to the options (A, B, C, etc.) between the 6th and 9th layers. Analyzing critical EEG channels for different tasks, we find that NeuroLM seems to aggregate information to Cz for most datasets. For HMC, F4 and C3 are crucial, while O2 is less effective in sleep stage classification.

F ANALYSIS OF NEURAL TOKENIZER

F.1 ABLATION ON TEMPORAL-FREQUENCY PREDICTION

Temporal and frequency domains are two pivotal aspects of EEG signals. To investigate the importance of these two domains for different downstream tasks, we study three variants by setting the reconstruction target in neural tokenizer training as only the temporal domain, only the frequency domain, and both temporal and frequency domains (original NeuroLM). Figure 10 shows the comparison between the three variants. Interestingly, it can be found that the temporal domain plays a more crucial role on TUAB, Workload, and TUSL. On the contrary, reconstructing the frequency components obtains better performance on TUEV, SEED, and HMC, indicating that the frequency domain is of great importance for event classification, emotion recognition, and sleep stage classification. By combining the two domains, most tasks achieve similar or higher performance, demonstrating the effectiveness of our neural tokenizer that excavates compact EEG representations for language models.

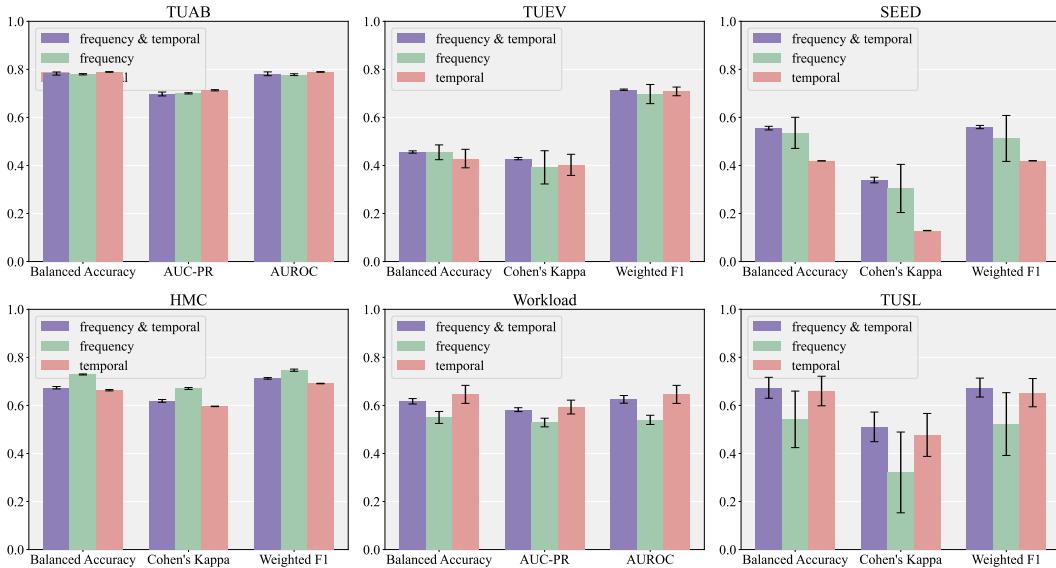


Figure 10: Ablation study on reconstructing temporal or frequency domain in neural tokenizer.

F.2 VISUALIZATION OF EEG AND TEXT EMBEDDINGS

To evaluate the effectiveness of EEG-text embedding space alignment, we visualize the embeddings in Figure 11 using t-SNE (Van der Maaten & Hinton, 2008). The EEG embeddings expand outside the text space without alignment. In this case, we find that the model fails to predict the answers we expect in multi-task instruction tuning, i.e., the model will output random words instead of options

like (A), (B), or (c) in choice questions. When training with alignment, the EEG space mostly aligns with text space, resulting in normal prediction in instruction tuning, proving the necessity of EEG-text alignment.

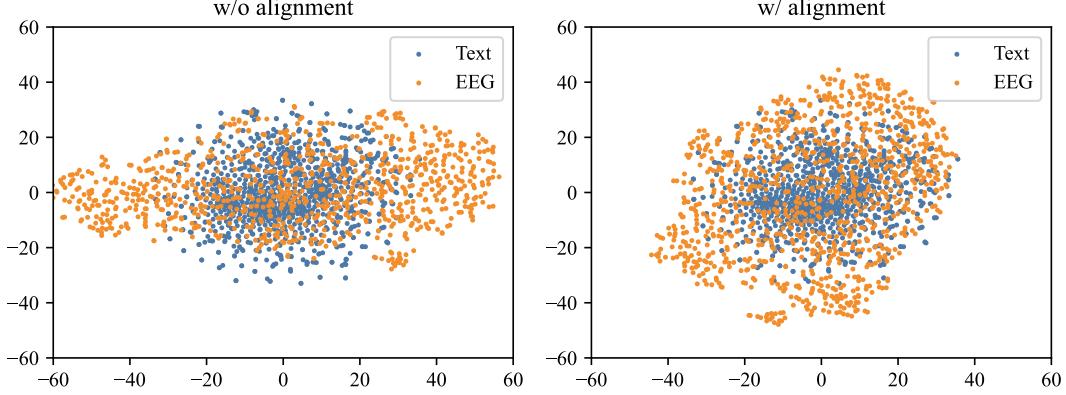


Figure 11: Representation visualization of EEG and text by t-SNE. **Left:** training neural tokenizer without alignment. **Right:** training neural tokenizer with alignment.

G ABLATION ON DIFFERENT PRE-TRAINING EPOCHS

We conduct an ablation study on tuning the pre-trained models from different epochs to testify the best pre-training epoch. As shown in Table 10 11 12, we use the pre-trained models of 5, 10, 15, and 20 epochs. Bold represents the best results and underline represents the second best results. It can be found that pre-training for 20 epochs obtains the most bold and underlined results. The performance of 5 epochs gets its best on SEED and HMC while the model from 10 epochs achieves the best result on Workload. Overall, pre-training for more epochs can lead to good performance in different tasks.

Table 10: Results on TUAB and TUEV.

Pre-trained Epochs	TUAB			TUEV		
	Balanced Acc.	AUC-PR	AUROC	Balanced Acc.	Cohen's Kappa	Weighted F1
5	0.7763±0.0037	0.6891±0.0040	0.7762±0.0038	0.4609±0.0696	0.3914±0.0670	0.6975±0.0249
10	0.7738±0.0040	0.6995 ±0.0038	0.7647±0.0143	0.4693 ±0.0175	0.4625 ±0.0091	0.7353 ±0.038
15	<u>0.7780</u> ±0.0050	0.6951±0.0054	0.7735±0.0112	0.4557±0.0277	0.4216±0.0215	0.7131±0.0101
20	0.7826 ±0.0065	0.6975±0.0081	0.7816 ±0.0079	0.4560±0.0048	0.4285±0.0048	0.7153±0.0028

Table 11: Results on SEED and HMC.

Pre-trained Epochs	SEED			HMC		
	Balanced Acc.	Cohen's Kappa	Weighted F1	Balanced Acc.	Cohen's Kappa	Weighted F1
5	0.5641 ±0.0103	0.3505 ±0.0172	0.5679 ±0.0123	0.6956 ±0.0136	0.6269 ±0.0053	<u>0.7118</u> ±0.0048
10	0.5553±0.0089	0.3376±0.0143	0.5592±0.0091	<u>0.6763</u> ±0.0054	0.6161±0.0069	0.7065±0.0057
15	0.5543±0.0156	0.3365±0.0244	0.5572±0.0164	0.6543±0.0151	0.5991±0.0153	0.6674±0.0265
20	<u>0.5554</u> ±0.0075	0.3393±0.0117	0.5599±0.0068	0.6737±0.0050	0.6188±0.0057	0.7126 ±0.0034

Table 12: Results on Workload and TUSL.

Pre-trained Epochs	Workload			TUSL		
	Balanced Acc.	AUC-PR	AUROC	Balanced Acc.	Cohen's Kappa	Weighted F1
5	0.5816±0.0235	0.5483±0.0160	0.5815±0.0236	0.5342±0.0235	0.2950±0.0345	0.5241±0.0260
10	0.6540 ±0.0192	0.6123 ±0.0165	0.6501 ±0.0178	<u>0.5920</u> ±0.0560	0.3884±0.0876	<u>0.5984</u> ±0.0574
15	0.5701±0.0282	0.5426±0.0177	0.5682±0.0255	0.5910±0.0629	<u>0.3915</u> ±0.0999	0.5868±0.0536
20	0.6172±0.0113	0.5824±0.0080	0.6253±0.0160	0.6734 ±0.0436	0.5107 ±0.0617	0.6743 ±0.0394

H DISCUSSION

Limitations. NeuroLM represents the first attempt to integrate various EEG downstream tasks into a unified model, achieving promising results across multiple downstream datasets. However, it has some limitations: 1) Although NeuroLM can surpass certain single-task baselines, it still lags behind state-of-the-art methods that are end-to-end trained on each downstream dataset. 2) NeuroLM is somewhat sensitive to hyperparameter settings, and may not yield satisfactory results without careful tuning. 3) With limited high-quality EEG-text pairs available, this paper only employs coarse-grained alignment between EEG and language, i.e., space-wise alignment, which can pose challenges for LLMs in extracting useful information from EEG tokens.

Outlook. Reflecting on the outlook part of the LaBraM paper, this work explores the first and third suggested directions. Looking ahead, we foresee several potential improvements: 1) Utilizing more advanced LLMs as the base models. While this paper uses GPT-2, a relatively small LLM, and still achieves promising results in the multi-task paradigm, leveraging newer, more advanced open-source LLMs such as LLaMA 3 (Dubey et al., 2024) may significantly enhance NeuroLM’s multi-task learning capabilities. 2) Adopting the mixture-of-experts approach is another promising direction. Given the modality gap between EEG and language, using modality-specific experts may improve multimodal learning with LLMs. 3) Developing finer-grained EEG and text alignment methods, such as describing EEG samples with predefined sentences and aligning EEG and text descriptions at the VQ training stage by adding a contrastive learning loss, may further enhance performance.

BELT-2: BOOTSTRAPPING EEG-TO-LANGUAGE REPRESENTATION ALIGNMENT FOR MULTI-TASK BRAIN DECODING

Jinzhao Zhou, Yiqun Duan, Fred Chang, Thomas Do, Yu-Kai Wang, Chin-Teng Lin

ABSTRACT

The remarkable success of large language models (LLMs) across various multi-modality applications is well established. However, integrating large language models with humans, or brain dynamics, remains relatively unexplored. In this paper, we introduce BELT-2, a pioneering multi-task model designed to enhance both encoding and decoding performance from EEG signals. To bolster the quality of the EEG encoder, BELT-2 is the first work to innovatively 1) adopt byte-pair encoding (BPE)-level EEG-language alignment and 2) integrate multi-task training and decoding in the EEG domain. Inspired by the idea of *Bridging the Brain with GPT*, we further connect the multi-task EEG encoder with LLMs by utilizing prefix-tuning on intermediary output from the EEG encoder. These innovative efforts make BELT-2 a pioneering breakthrough, making it the first work in the field capable of decoding coherent and readable sentences from non-invasive brain signals. Our experiments highlight significant advancements over prior techniques in both quantitative and qualitative measures, achieving a decoding performance with a BLEU-1 score of 52.2% on the ZuCo dataset. Furthermore, BELT-2 shows a remarkable improvement ranging from 31% to 162% on other translation benchmarks. Codes can be accessed via the provided anonymous link¹.

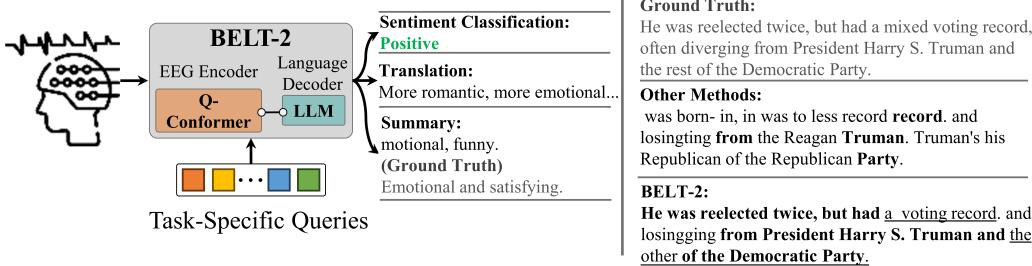


Figure 1: Overview of BELT-2. The first work of multi-task brain decoding by bridging the Q-Conformer EEG encoder and LLMs. Provided samples also suggest BELT-2 is the first to achieve fluent sentence decoding results from noninvasive brain signals.

1 INTRODUCTION

Recently, the emergence of large language models (LLMs) has spurred efforts to integrate them with various modalities, such as VisualLLMs (Liu et al., 2023b; Oquab et al., 2023), and Robotics (Driess et al., 2023). These methods achieved remarkable improvement in various task settings. Yet, an important topic, the direct combination of LLMs with human intention remains relatively unexplored. Nonetheless, the inherent subject-wise non-stationary characteristics of Electroencephalography (EEG) signals, coupled with rigorous experimental protocols, make the task of decoding words or sentences exceptionally challenging.

¹<https://anonymous.4open.science/r/BELT-2-0048>

Explorations on brain-to-text and brain-to-speech decoding in the earlier stage (Herff et al., 2015; Makin et al., 2020; Panachakel & Ramakrishnan, 2021; Nieto et al., 2021) mostly perform decoding on a closed word-level set, which still has notable restrictions on vocabulary size and limitations to more intricate application scenarios. For the brain-to-language decoding, EEG-to-Text (Wang & Ji, 2022) introduced the open-vocabulary decoding of EEG signals with an initial performance baseline. DeWave (Duan et al., 2023) improved decoding performance by introducing a discrete encoder for EEG. BELT (Zhou et al., 2023a) which boosted decoding performance by leveraging language supervision. However, these methods are limited to single-task settings and have not achieved multi-task decoding from brain signals to natural languages. An extensive **related works** is provided in Appendix A due to space limit.

In this paper, we propose BELT-2, the first EEG-language learning framework to bridge the modality gap and effectively exploit LLM’s generative capacity for EEG decoding. BELT-2 enhances three key aspects of brain decoding research. 1) It is the first to introduce **BPE-level contrastive learning** for EEG-to-language alignment. 2) It first introduces a **prompt-based multi-task encoder** for EEG research. 3) It proposes a cost-effective solution for connecting an EEG encoder with a large language model (LLM).

More specifically, we introduce a novel discrete querying conformer (Q-Conformer) as the EEG encoder to improve encoding capacity and enable multitasking (Figure 5). Unlike previous single-task EEG encoders (Zhou et al., 2023a; Duan et al., 2023), Q-Conformer is able to extract task-specific contexts according to a given query prompt. For the training of Q-Conformer, we propose the BPE-level EEG-language contrastive learning (BPE-CL) to bootstrap the learning of language-aligned EEG representation. After training, we bridge the Q-Conformer and an LLM decoder by prefix-tuning with both models frozen. To improve the performance of the bridging, we further propose a technique called speculative augmentation (SA) to improve the training efficiency. The main contributions of BELT-2 could be concluded in four aspects.

- This paper presents a novel framework capable of decoding fluent open-vocabulary sentences, facilitating multi-task EEG decoding including EEG translation, sentiment classification, and summarization.
- The Q-Conformer is proposed to improve the encoding ability and the scalability for multi-tasking while the BPE-level contrastive learning establishes a firm alignment between EEG and language representations.
- This paper provides a cost-effective bridging method for connecting LLMs with brain encodings by turning virtual-prefix. A speculative augmentation method is introduced to further improve the bridging performance.
- Experimental results suggest that the proposed BELT-2 exceeds SOTA performance on different EEG decoding tasks. For EEG translation, BELT-2 achieves 52.59 BLEU-1, 17.85 BLEU-4, and 40.1 Rouge-1 Precision, which significantly outperforms the previous baseline by 31%, 162% and 26% respectively. On sentiment classification, BELT-2 achieves 74.62% accuracy without further assistance from additional classifiers or external datasets. BELT-2 is also the first work that achieves EEG summarization with a SOTA 31.17 BLEU-1 score.

2 BELT-2

BELT-2 introduces the Q-Conformer which enhances both the capacity to encode EEG information and the extendibility to multi-task. To bridge the modality gap between EEG and language, we boost EEG-to-Language representation learning through two learning stages: (1) the EEG-to-language alignment learning stage for learning the Q-Conformer EEG encoder. (2) a prefix-tuning stage for bridging Q-Conformer with LLM.

2.1 Q-CONFORMER AS EEG ENCODER

The overall structure of the Q-Conformer is illustrated in Figure 5 which consists of a discrete conformer, a Context Transformer (C-Former), and a query prompt. The discrete conformer functions as a discrete EEG tokenizer that captures primitive patterns from the input EEG embeddings. The C-Former extracts mid-layer coding (MLC) that contains context information specific to a given task given by the learnable query prompt.

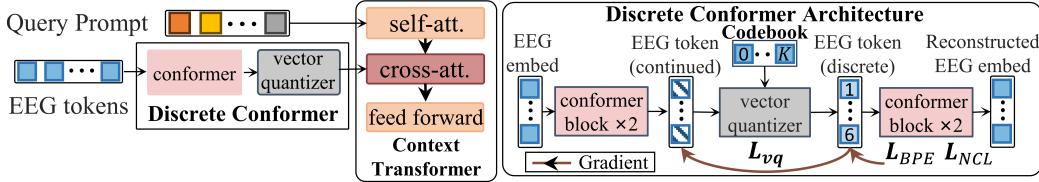


Figure 2: The overall structure of the Q-Conformer. It consists of a discrete conformer, a context transformer (C-Former), and a query prompt. The input EEG embeddings (EEG embed) are first processed by the conformer into continuous EEG tokens. A vector quantizer is then used to discretize the EEG tokens. Then, a query prompt interacts with the discrete EEG token via the cross-attention layer from in the C-Former to extract task-specific context information from the discrete EEG tokens.

Discrete Conformer: The discrete conformer consists of a conformer model and a vector quantizer. After preprocessing, the raw EEG waveform is segmented into windows using eye-tracking information. Then a frequency domain transform converts EEG segments into fix-size EEG embeddings $\mathbf{e} \in \mathbb{R}^{L \times N \times D}$. L is the maximum length of the embedding sequence, N denotes the number of EEG channels, and D denotes the embedding size. The conformer model consists of 2 conformer blocks which follow the structure manifested in (Gulati et al., 2020). The conformer model $E(\cdot)$ converts the EEG embeddings \mathbf{e} into continuous EEG tokens $\mathbf{h} \in \mathbb{R}^{L \times N \times d}$, where d denotes the size of the continuous EEG tokens.

We then convert \mathbf{h} to a set of discrete tokens \mathbf{b} by a vector quantizer (VQ) that looks up the nearest discrete code \mathbf{v}_k , $k = \{0, 1, \dots, K\}$ from the codebook \mathcal{V} (Razavi et al., 2019). The quantization process $\mathbf{z}_q(\mathbf{h})$ can be written as Equation 1.

$$\mathbf{z}_q(\mathbf{h}) = \{\mathbf{z}_q(\mathbf{h}_i)\}_{i=0}^L, \quad \mathbf{z}_q(\mathbf{h}_i) = \mathbf{v}_k, \quad k = \arg \min_j \|\mathbf{h}_j - \mathbf{v}_j\|_2^2 \quad (1)$$

We use L_{vq} (Equation 2) to train the discrete codebook. The L_{vq} is a weighted summation of 4 loss terms. The first two terms are the codebook loss and the commitment loss. They are used to update the codebook by minimizing the information loss between the input and the output discrete tokens Van Den Oord et al. (2017). The third term encourages the balanced use of all entries in the codebook and prevents codebook collapse during training (Dieleman et al., 2018). The last term is a reconstructive loss that ensures the information passed to the VQ is sufficient to describe the EEG signal.

$$\mathcal{L}_{vq} = \|\text{sg}[\mathbf{h}] - \mathbf{z}_q(\mathbf{h})\|_2^2 + \|\mathbf{h} - \text{sg}[\mathbf{z}_q(\mathbf{h})]\|_2^2 + \frac{1}{|\mathcal{V}|} \sum_{k=0}^{|\mathcal{V}|} p_k \log p_k + \|\mathbf{e} - \hat{\mathbf{e}}\|_2^2 \quad (2)$$

, where $\text{sg}[\cdot]$ stands for the stop-gradient operator which is an identity at the forward pass while having zero gradients during the backward pass. $|\mathcal{V}|$ denotes the size of the discrete codebook and p_k denotes the softmax probability of the codebook entry k being used in each batch. $\hat{\mathbf{e}}$ denotes the reconstructed EEG embedding from $\mathbf{z}_q(\mathbf{h})$ using 2 conformer blocks.

C-Former and Query Prompt We create a set number of learnable query embeddings (query prompt) as input to the C-Former. The C-Former is composed of self-attention layers and cross-attention layers arranged in consecutive order. After feeding the query prompts and the discrete EEG tokens into the C-Former, the query prompts interact with each other through the self-attention layers and further interact with the discrete EEG tokens through the following cross-attention layer. A new query prompt will be initialized when training the Q-Conformer for a specific task. After training on a specific task, the query prompts learn to act as the instruction of the current task that guides the C-Former to extract MLC as the task-specific context from the EEG modality.

This querying mechanism enables a more flexible adaptation of the pretrained Q-Conformer to a new downstream task by adding a new set of query prompts. It also allows the reusing of knowledge learned from previous training tasks. In our experiment setup, we initialize the C-Former with the pre-trained weights of BART_{large} (Lewis et al., 2019). We employ a query prompt of 20 learnable tokens for a specific, with each query possessing a dimensionality of 1024.

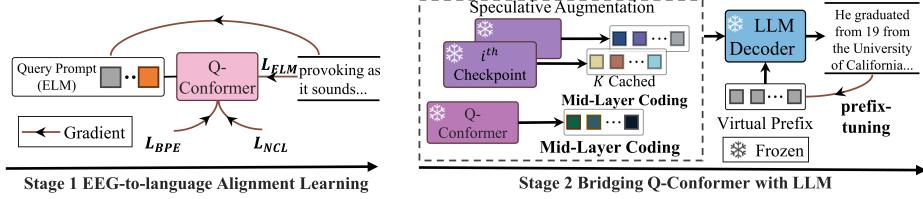


Figure 3: BELT-2’s two-stage training schema. For EEG-to-language alignment learning (**left**), we jointly optimize three objectives that firmly establish the EEG-to-language alignment and enforce the query prompt to extract the EEG context most relevant to a task. For bridging of Q-Conformer and LLM (**right**), connect a frozen EEG model (Q-Conformer) and a frozen LLM by tuning the continuous virtual prefix using the prefix-tuning method. Speculative augmentation is used to boost the performance of the prefix-tuning process.

2.2 EEG-TO-LANGUAGE ALIGNMENT LEARNING

In the EEG-to-language alignment learning stage, we train the Q-Conformer and align the encoded EEG tokens to the language modality. To achieve EEG-to-Language alignment, we combine two contrastive objectives and a pretraining objective to the VQ objective in Equation 2. The two contrastive objectives include (1) BPE-level contrastive learning (BPE-CL), and (2) Negative Contrastive learning (NCL). We further pretrain the Q-Conformer to achieve a task-specific query prompt by the EEG-to-Language matching (ELM) objective, which guides the C-Former to extract MLC that contains the most relevant EEG contexts in the specific task.

BPE-level contrastive learning (BPE-CL) learns to align the discrete EEG tokens with BPE subword embeddings by maximizing their mutual information. Unlike the BELT-1 model (Zhou et al., 2023a) where contrastive learning is only performed at the word level, we perform EEG-language alignment in the BPE subword level to improve EEG-language alignment. Given the limited size of EEG-language pairs in the training set, this method enforces stronger semantic guidance to the EEG representation while enhancing the matching of subword units that are out-of-training vocabulary.

The sampling strategy of the BPE-CL is illustrated in Figure 4. We commence by converting words into BPE tokens $w \in \mathcal{W}$, e.g., converting “Visually” into [“Vis”, “ually”]. The embeddings of these BPE tokens serve as positive targets for the EEG token corresponding to “Visually” while BPE tokens other words are viewed as negative targets. We uniformly sample 1 positive target and K negative targets for each discrete EEG token in a training batch. The learning objective L_{bpe} for the discrete EEG tokens and the BPE embeddings is formulated as:

$$\mathcal{L}_{bpe} = -\log \frac{\exp(\mathbf{z}_q(\mathbf{h})^\top \mathbf{w}^+)}{\exp(\mathbf{z}_q(\mathbf{h})^\top \mathbf{w}^+) + \sum_{i=1}^K \exp(\mathbf{z}_q(\mathbf{h})^\top \mathbf{w}^-)}, \quad (3)$$

, where \mathbf{w}^+ is the sampled embedding of the positive BPE token and \mathbf{w}^- is the negative ones.

Negative contrastive learning (NCL) aims to further improve the distinctions between the discrete EEG tokens by randomly sampling K negative EEG tokens as distractors for each discrete EEG token in a training batch, which is defined as:

$$\mathcal{L}_{neg} = -\log \frac{1}{\sum_{i=1}^K \exp(\mathbf{z}_q(\mathbf{h})^\top \mathbf{z}_q(\mathbf{h})^-)}, \quad (4)$$

, where $\mathbf{z}_q(\mathbf{h})^-$ are sampled negative tokens from the batch and $\mathbf{z}_q(\mathbf{h})$ is defined in Equation 1. This objective enlarges the distinction among EEG tokens that are indistinguishable upon reading different words, easing the decoding effort.

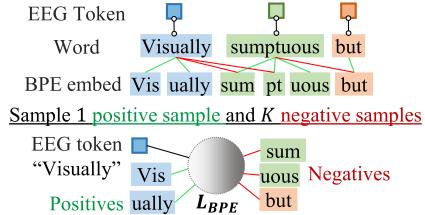


Figure 4: The illustration of BPE-level contrastive learning.

EEG-to-language matching (ELM) aims to function as the pretraining task for learning the initial task-specific query prompt, which in terms is used to instruct the C-Former to extract task-specific context from the EEG tokens. We use a sequence-to-sequence machine translation loss similar to previous works Zhou et al. (2023a); Wang & Ji (2022); Duan et al. (2023) as the objective function. Given the word-level EEG embedding sequence and text sentence pair $\langle \mathcal{E}, \mathcal{S} \rangle$, we maximize the probability of the decoded sentence $p(\mathcal{S}|\mathcal{E})$ produced by the Q-Conformer. The learning objective is a machine translation term L_{tr} , which could be written as follows:

$$\mathcal{L}_{elm} = - \sum_l^L \log p(s_l \in \mathcal{S} | \mathbf{q}) \quad (5)$$

, where L is the total length of the target text sequence, $s_l \in \mathcal{S}$ denotes the decoded tokens from the C-Former and \mathbf{q} denotes the query prompt.

2.3 BRIDGING Q-CONFORMER WITH LLM

We propose to bridge the frozen Q-Conformer and a frozen LLM to leverage both models effectively for EEG-to-Language tasks by tuning a set of virtual prefixes added to the output embeddings of the Q-Conformer, in order to achieve stronger performance at a lower training cost.

Prefix-tuning To achieve a proper prefix prompt that can steer the LLM to decode the MLC without changing the LLM’s parameters, we adopt the prefix-tuning (Li & Liang, 2021) method to only train a set of virtual prefix tokens as prompts to the LLM. In particular, we concat the virtual prefix and the MLC from the Q-Conformer as input to the subsequent frozen LLM. Please refer to Appendix C.3 for more details on prefix-tuning.

Speculative Augmentation (SA) Despite the use of the lightweight prefix-tuning method, the size and diversity of training samples are still lacking. This is because while the Q-Conformer learns to extract task-specific context, it also learns to ignore task-irrelevant information. This would be a well-anticipated perk for an EEG encoder if we choose to directly decode language output from the EEG encoder. However, it also significantly reduces the diversity of training samples, making the learning of a good prefix difficult.

Our BELT-2 framework solves this issue by proposing the SA method to sample MLC from a total of $K+1$ Q-Conformer checkpoints to provide more diverse prefix-tuning samples. In particular, we randomly sample K model checkpoints other than the best-performing checkpoint to produce MLC for the prefix-tuning. During the forward process, a speculative ratio r is defined to determine whether to use best checkpoint or one of the K suboptimal checkpoints. To reduce the cost of memory, we cache the output MLCs of these K model checkpoints during the training of Q-Conformer to avoid actually loading the checkpoints in the prefix-tuning stage.

In our experiment, we set $K = 15$ for a balance of performance and training costs to achieve a $6\times$ larger and more diverse training sample set for the tuning of the LLM Decoder.

2.4 EXTENDING DECODING TO MULTI-TASK

Translation: Our definition of the EEG-to-Text translation task follows previous works on this topic (Wang & Ji, 2022). Given the word-level EEG embedding sequence and text sentence pair $\langle \mathcal{E}, \mathcal{S} \rangle$, we maximize the probability of the decoded sentence $p(\mathcal{S}|\mathcal{E})$ produced by our model. The training objective L_{tr} for the translation task could be written as follows:

$$p(\mathcal{S}|\mathcal{E}) = \prod_{l=1}^L p(s_l|\mathcal{E}, s_{<l}), \quad \mathcal{L}_{tr} = - \sum_l^L \log p(s_l \in \mathcal{S}) \quad (6)$$

where L is the total length of the target text sequence and $s_l \in \mathcal{S}$ denotes the word tokens produced by our model.

Summary: We propose the first EEG-to-text summarization task by creating a summary dataset from the Zuco datasets. Human attention lingers around keywords and pivotal concepts during reading (Ding et al., 2022). Consequently, we hypothesize that the extraction of key concepts could be a more direct way to facilitate the transmission of neural information and the understanding of a

person’s intention. As such, our nuanced summarization task not only enhances our understanding of EEG data but also opens up exciting possibilities for advancing research in cognitive science.

We kickstart by constructing the prompt “*Rewrite the sentence by summarizing its main idea using {T} words from the sentence, and keep the summarized sentence similar to the original sentence: {s}*” with $\{s\}$ being each ground truth sentence from the ZuCo dataset and attain the initial summarization targets for each sentence. We set $T = 8$ in our experiment and use the LLAMA2 model (Touvron et al., 2023) to generate the initial summarization targets. Afterwards, manual inspection and rectification are carried out to improve the dataset’s reliability and informativeness. The word-level EEG embedding sequence and summary pair are denoted by $\langle \mathcal{E}, \hat{\mathcal{S}} \rangle$. To extend the Q-Conformer for summarization task, a new query prompt for summarization will be added. The training objective for generating summaries is similar to Equation 6, with the sole alteration being the substitution of \mathcal{S} with $\hat{\mathcal{S}}$. For multi-task training, we train all tasks simultaneously by randomly sampling tasks for each update iteration.

Sentiment Classification: We could further extend the Q-conformer to perform the sentiment classification task by adding another query prompt for the Q-Conformer and using the last output token from the Q-conformer as the CLS token. In particular, we use the EEG-sentiment label pair $\langle \mathcal{E}, c \rangle$. Unlike Wang & Ji (2022), we don’t need to use external sentiment classification datasets or learn an additional classifier. The training objective for sentiment classification is as follows:

$$\mathcal{L}_{st} = - \sum_{i=1}^{|C|} c_i \log p(\hat{c} | \mathcal{E}_i), \quad (7)$$

, where $|C|$ is the number of the sentiment categories and \hat{c} is the sentiment prediction.

3 EXPERIMENT AND RESULTS

3.1 EXPERIMENT SETUP AND IMPLEMENTATION DETAILS

We use the ZuCo datasets (Hollenstein et al., 2018; 2019) for the training and evaluation of the proposed BELT-2 framework. The ZuCo datasets contain EEG data recorded during natural reading tasks with eye-tracking data for word-level EEG segmentation. Reading material is collected from movie reviews (Socher et al., 2013) and Wikipedia articles. We split the dataset into train, val, and test subsets (80%, 10%, 10%). In this cross-sentence setting, sentences will not overlap among any two subsets. In addition, cross-subject performance is also evaluated. We evaluate translation and summary performance using the BLEU scores (Papineni et al., 2002) and ROUGE-1 scores Lin (2004). We use **P**, **R**, **F1**, and **Acc** to denote precision, recall, F1-score, and accuracy respectively.

3.2 IMPLEMENTATION DETAILS

The code could be assessed through an anonymous link². For the word-level EEG embeddings, the total length of an embedding sequence is $L = 56$ and the embedding size is $d = 840$. The discrete conformer has 8 attention heads with the feed-forward dimension size of 2048 and a discrete codebook with 1024 entries with a latent size of 1024. The number of querying tokens used for The Q-Conformer is 20. We train the Q-Conformer with a learning rate of $5e^{-06}$ for 60 epochs during EEG-to-language alignment learning using AdamW (Loshchilov & Hutter, 2017). For the bridging stage, we use 8 virtual prefix and set the speculative augmentation factor K to 15 with a speculative ratio of 0.3. We use pre-trained BART and T5 models from the huggingface platform to initialize the Q-conformer and the LLM decoder. We also conducted experiments of massive size LLAMA2 model³ in Section 3.5. Due to the limitation of space, refer to Appendix C for more details.

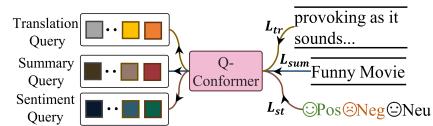


Figure 5: For multi-task training, we train three tasks simultaneously by randomly sampling tasks for each training iteration. Each task-specific query prompt learns to provide task-specific instructions by training on the corresponding task-specific objective function.

²<https://anonymous.4open.science/r/BELT-2-0048>

³<https://huggingface.co/meta-llama/Llama-2-7b>

3.3 TRANSLATION PERFORMANCE

Quantitative Results We show quantitative results in Table 1. Compared to previous methods, e.g., EEG-to-Text (Wang & Ji, 2022), Dewave (Duan et al., 2023), and BELT-1 (Zhou et al., 2023a) When only using EEG Encoder, We observe that the introduction of BPE-level contrastive learning bootstrapped a significant improvement (row 4 compared to row 5), achieving the SOTA EEG decoding BLEU-{1, 2, 3, 4} scores of 43.06, 25.57, 15.16, and 9.17, which outperform DeWave by 1.71, 1.42, 1.24, and 0.95. By further connecting with the LLM decoder, BELT-2 further achieves the BLEU-{1, 2, 3, 4} scores of 52.59, 36.32, 25.21, and 17.85, which brings additional 9.66, 10.96, 10.16, and 8.76 BLEU score improvements. The increase of the metrics is more significant for longer phrases (+162% for 4-gram and +99% for 3-gram) compared to the baseline EEG-to-Text method. Additionally, we present ablation results that analyze the influence of VQ and the BPE-CL within our model, revealing that the utilization of BPE-CL significantly contributes to the enhancement of performance. However, multitask training did not bring a significant improvement to the translation result, which is elaborated in the Appendix F.

Table 1: Quantitative Results on Brain-to-Language Translation on the ZuCo Datasets.

Model	Vector Quantizer	BPE-CL	Enable Multi-Task	Prefix Tuning	BLEU-N (%)				ROUGE-1 (%)		
					N=1	N=2	N=3	N=4	R.	P.	F1
EEG-to-Text	✗	✗	✗	✗	40.12	23.18	12.61	6.80	28.8	31.7	30.1
Dewave	✓	✗	✗	✗	43.35	24.15	13.92	8.22	28.82	33.71	30.69
BELT-1	✓	✗	✗	✗	42.31	25.26	14.81	8.73	29.86	36.06	32.57
BELT-2	✓	✓	✓	✗	43.06	25.57	15.05	9.09	30.28	34.12	31.99
BELT-2+LLM(T5)	✓	✓	✓	✓	52.38	36.28	25.28	17.95	36.08	39.47	37.59
BELT-2 Ablations											
BELT-2	✓	✗	✓	✓	41.57	24.02	13.80	8.06	29.35	32.46	30.74
BELT-2	✗	✓	✓	✗	41.90	24.57	14.2	8.28	29.60	34.03	31.54

Table 2: Qualitative results on unseen EEG signals. The **bold** denotes an exact match between the ground truth and our prediction. underline denotes a fuzzy match with similar semantic meanings.

(1)	Target	He is a prominent <u>member</u> of the Bush family, the younger brother of President George W. Bush and the second son of former President George H. W. Bush and Barbara Bush.
	Others	was a former member of the American <u>family</u> , and first <u>brother</u> of President George W. Bush.
	Ours	the father <u>son</u> of President President George H. W. Bush. his Bush.
	Target	He was great member <u>member</u> of the American <u>family</u> , and <u>younger</u> brother of President George H. Bush
	Ours	and the younger cousin of President President George H. W. Bush. the Bush.
(2)	Target	Adolf Otto Reinhold Windaus (December 25, 1876 - June 9, 1959) was a significant German chemist.
	Others	rian Hitler,hardt,eren18 18, 1885 – January 3, 18) was a German figure- and
	Ours	Adolf Hitlero vonhard voner (J 15, 1875 - January 15, 1945) was a German German industrialpacist
(3)	Target	It just doesn't have much else... especially in a moral <u>sense</u> .
	Others	was so's work the to and not the country <u>sense</u> .
	Ours	It just doesn't work the of going except in the a way <u>sense</u> .
(4)	Target	He was reelected twice, but had a mixed <u>voting record</u> , often diverging from President Harry S. Truman and the rest of the Democratic Party.
	Others	was a- in, never to less record <u>record</u> , and losintg from his Reagan Truman . Truman's his Republican of the Republican Party.
	Ours	He was reelected twice, but had <u>voting record</u> , and losintg from President Harry S. Truman and the other of the Democratic Party.
(5)	Target	Following the 1980 presidential election, Bush and his family moved to Miami-Dade County, Florida.
	Others	the deaths <u>election</u> , the was his wife moved to California, Dade County, Florida.
	Ours	After his election <u>presidential election</u> , Reagan and his family moved to Miami,Dade County, Florida.

Cross-Subject Results As cross-subject performance is of vital importance for practical usage, we further report translation performance in cross-subject settings where we leave one subject out for evaluation and train the model using other subjects. Figure 6 shows the cross-subject translation performance for a total of 10 subjects compared to the cross-sentence result we achieved in the cross-sentence setting (Table 1). The radar charts in Figure 6 denote the performance is stable across different subjects with subjects achieving BLEU-1 scores ranging from 48.04 to 51.41.

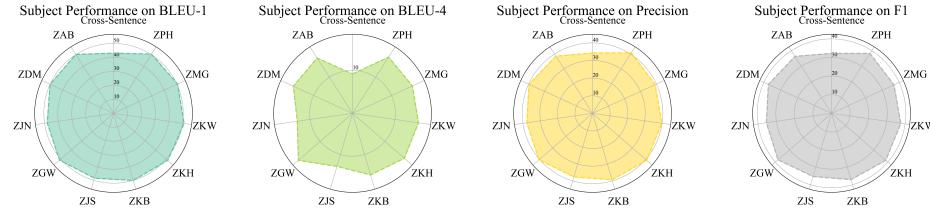


Figure 6: The cross-subjects performance for translation task.

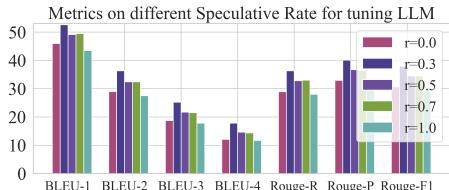


Figure 7: Ablation on speculative ratio.

Model	BLEU (%)		Rouge-1		
	N=1	N=3	P.	R.	F1
EEG-to-Text	25.14	0	10.37	7.30	8.49
BELT-2 w/o Pretrained	26.87	2.08	9.84	11.06	10.34
BELT-2 w/ Pretrained	31.17	5.09	12.73	13.26	12.91

Table 3: Quantitative Results of Summary Task

Qualitative Evaluation We showcase the generated text alongside the established approach from Wang & Ji (2022) in Table 2. We observe that BELT-2 generates more fluent sentences with greater grammatical coherence. Notably, our model adeptly captures subject-predicate relationships while other methods miss the subject and predicate. This is demonstrated by the accurate decoding of phrases like “*He was*” vs. “*He is*”, “*It just doesn’t work*” vs. “*It just doesn’t have*”. Furthermore, for sentence structures involving quoted dates, such as “(January 15, 1875 - January 15, 1945)” vs. “(December 25, 1876 - June 9, 1959)”, were also consistently deciphered.

3.4 MULTI-TASK PERFORMANCE

Sentiment Classification As shown in Table 4, previous works need to train an LLM classifier using an external Stanford Sentiment Treebank dataset (around 11,000 sentences) (Socher et al., 2013) and a new EEG encoder due to poor performance when training directly on the ZoCo dataset (Row 1-3). In contrast, an EEG encoder incorporating external classifiers (row 4-7) demonstrated improved performance (Wang & Ji, 2022). Our proposed Q-Conformer Encoder, achieve the state-of-the-art sentiment classification accuracy of 74.62% on the ZuCo dataset. We also observe that our method could effectively leverage pretrained knowledge from the translation task to improve performance (row 8-9).

Summarization We compare the summarization performance of the BELT-2 model with the EEG-to-Text model as the baseline. As shown in Table 3, the EEG-to-Text struggles to generate summarization while the proposed BELT-2 model exhibited better generative capacity, especially in longer phrases. Compared to using a newly initialized encoder (row 2), our BELT-2 exhibits a remarkable capacity to utilize the pretrained knowledge to increase the performance for the summarization task (row 3). Generally, it attains the BLEU-{1, 2, 3, 4} scores of 31.17, 15.7, 8.91, 5.09, outperforming the baseline method.

3.5 ABLATION STUDY

Bridging Q-Conformer Encoder with different LLMs Table 1 shows the result of bridging our Q-Conformer encoder with the T5 (Raffel et al., 2020). In Table 5, we conduct a comprehensive investigation of bridging LLM decoders with the Q-Conformer model, including the LLAMA2, T5, and the PEGASUS (Zhang et al., 2020) models. Results show that T5 LLMs consistently outperform other variants and boost the decoding performance. We attribute this superiority to T5’s denoising training objectives. However, the sheer scale of the LLM decoder does not necessarily lead to enhanced decoding performance. For example, PEGASUS and LLAMA2 did not yield much improvement in the translation performance.

Table 4: Quantitative Results of Sentiment Classification

EEG Encoder	Additional CLS Model	Additional Dataset	Acc.	P.	R.	F1
MLP	None	None	31.8	32.8	33.6	27.5
Bi-LSTM	None	None	30.9	27.5	33.6	17.4
Transformer	BERT	None	36.6	23.7	34.5	27.2
EEG2Text	BART	SST	55.30	62.40	56.50	55.60
BELT-1	BART	SST	65.13	63.67	63.34	62.45
BELT-1	Albertv2	SST	60.09	61.63	60.03	59.56
BELT-1	XLNet	SST	67.32	66.55	65.71	65.02
BELT-2 w/o Pretrained	None	None	59.74	57.67	57.63	57.11
BELT-2 w/ Pretrained	None	None	74.62	75.34	73.84	73.31

Table 5: Ablation study of bridging Q-Conformer Encoder with different LLMs

LLM	Type	BLEU-N (%)				ROUGE-1 (%)		
		N=1	N=2	N=3	N=4	P.	R.	F1
LLAMA2	7B	21.40	6.96	3.38	2.21	12.23	13.20	12.61
PEGASUS	google/pegasus-x-base	37.67	18.90	9.68	5.21	26.43	31.06	28.38
	google/pegasus-xsum	40.82	23.70	13.39	7.61	30.25	33.94	31.86
T5	t5-small	51.02	33.44	22.41	15.42	34.91	37.80	36.15
	t5-base	51.36	33.75	22.74	15.63	35.09	38.19	36.41
	t5-large	52.59	36.32	25.21	17.85	36.32	40.10	38.00
	google/flan-t5-base	50.01	33.09	21.77	14.49	32.97	36.64	34.54
	google/flan-t5-large	49.85	33.08	22.07	14.84	33.11	36.61	34.59

Speculative Augmentation We further conduct ablation experiments on the effect of different speculative ratios in Figure 7. We observe that the introduction of speculative augmentation at $r = 0.3$ has a significantly better impact on the decoding performance across all evaluated metrics.

LIMITATIONS

While BELT-2 achieved remarkable translation improvements by combining Q-Conformer with LLMs, it is worth noting that the accuracy still lags behind traditional language-to-language translation. Also, it is noted that the experiments were conducted on publicly available neural reading datasets with the help of eye-tracking markers. As a result, BELT-2 has not realized everyday communication such as ‘silent speech’ or ‘reading mind’. The vision of communication or controlling devices directly from brain dynamics remains a challenging task for follow-up research.

4 CONCLUSION

This paper introduces BELT-2, a pioneering EEG-language learning framework for bridging brain signals to LLMs. Our framework achieves EEG-to-language alignment by incorporating the novel BPE-CL objective and proposed an effective method for bridging a frozen Q-Conformer EEG Encoder and a frozen LLM to leverage their generative capacity. The multi-task extendibility of the Q-Conformer also establishes BELT-2 as the first work to achieve a multi-task decoding model in EEG research. Extensive experiments were conducted to evaluate the performance of BELT-2 quantitatively and qualitatively. Especially, this work provides the first study investigating the feasibility of using frozen pretrained LLM to process EEG contexts exemplified by a wide range of LLMs. Our experimental result shows that the BELT-2 framework represents a significant step forward in integrating human brain signals with LLMs, opening up exciting new avenues for research and development in cognitive neuroscience and brain-computer interfaces. We hope that this work will inspire further exploration and innovation in this exciting and rapidly evolving field.

REFERENCES

- Gopala K Anumanchipalli, Josh Chartier, and Edward F Chang. Speech synthesis from neural decoding of spoken sentences. *Nature*, 568(7753):493–498, 2019.
- Alan Cruttenden. *Gimson’s pronunciation of English*. Routledge, 2014.
- Karan Desai and Justin Johnson. Virtex: Learning visual representations from textual annotations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11162–11173, 2021.
- Sander Dieleman, Aaron van den Oord, and Karen Simonyan. The challenge of realistic music generation: modelling raw audio at scale. *Advances in neural information processing systems*, 31, 2018.
- Xiao Ding, Bowen Chen, Li Du, Bing Qin, and Ting Liu. Cogbert: Cognition-guided pre-trained language models. In *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 3210–3225, 2022.
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.
- Yiqun Duan, Jinzhao Zhou, Zhen Wang, Yu-Kai Wang, and Chin-Teng Lin. Dewave: Discrete eeg waves encoding for brain dynamics to text translation. *arXiv preprint arXiv:2309.14030*, 2023.
- Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.
- Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. Conformer: Convolution-augmented transformer for speech recognition. *arXiv preprint arXiv:2005.08100*, 2020.
- Christian Herff, Dominic Heger, Adriana De Pesters, Dominic Telaar, Peter Brunner, Gerwin Schalk, and Tanja Schultz. Brain-to-text: decoding spoken phrases from phone representations in the brain. *Frontiers in neuroscience*, 9:217, 2015.
- Nora Hollenstein, Jonathan Rotsztejn, Marius Troendle, Andreas Pedroni, Ce Zhang, and Nicolas Langer. Zuco, a simultaneous eeg and eye-tracking resource for natural sentence reading. *Scientific data*, 5(1):1–13, 2018.
- Nora Hollenstein, Marius Troendle, Ce Zhang, and Nicolas Langer. Zuco 2.0: A dataset of physiological recordings during natural reading and annotation. *arXiv preprint arXiv:1912.00903*, 2019.
- Armand Joulin, Laurens Van Der Maaten, Allan Jabri, and Nicolas Vasilache. Learning visual features from large weakly supervised data. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, pp. 67–84. Springer, 2016.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.
- Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023.

- Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. [arXiv preprint arXiv:2101.00190](#), 2021.
- Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In [Text summarization branches out](#), pp. 74–81, 2004.
- Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. [arXiv preprint arXiv:2301.12503](#), 2023a.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. [arXiv preprint arXiv:2304.08485](#), 2023b.
- Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. [arXiv preprint arXiv:1711.05101](#), 2017.
- Joseph G Makin, David A Moses, and Edward F Chang. Machine translation of cortical activity to text with an encoder–decoder framework. [Nature neuroscience](#), 23(4):575–582, 2020.
- Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. [arXiv preprint arXiv:2111.09734](#), 2021.
- David A. Moses, Sean L. Metzger, Jessie R. Liu, Gopala K. Anumanchipalli, Joseph G. Makin, Pengfei F. Sun, Josh Chartier, Maximilian E. Dougherty, Patricia M. Liu, Gary M. Abrams, Adelyn Tu-Chan, Karunesh Ganguly, and Edward F. Chang. Neuroprosthesis for decoding speech in a paralyzed person with anarthria. [New England Journal of Medicine](#), 385(3):217–227, 2021. doi: 10.1056/NEJMoa2027540. URL <https://doi.org/10.1056/NEJMoa2027540>.
- Nicolas Nieto, Victoria Peterson, Hugo Leonardo Rufiner, Juan Kamienskiowski, and Ruben Spies. ”thinking out loud”: an open-access eeg-based bci dataset for inner speech recognition. [bioRxiv](#), 2021.
- Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. [arXiv preprint arXiv:2304.07193](#), 2023.
- Jerrin Thomas Panachakel and Angara Ganesh Ramakrishnan. Decoding covert speech from eeg—a comprehensive review. [Frontiers in Neuroscience](#), 15:392, 2021.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In [Proceedings of the 40th annual meeting of the Association for Computational Linguistics](#), pp. 311–318, 2002.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In [International conference on machine learning](#), pp. 8748–8763. PMLR, 2021.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. [The Journal of Machine Learning Research](#), 21(1):5485–5551, 2020.
- Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. [Advances in neural information processing systems](#), 32, 2019.
- Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In [Proceedings of the IEEE/CVF conference on computer vision and pattern recognition](#), pp. 10684–10695, 2022.
- Prajwal Singh, Pankaj Pandey, Krishna Miyapuram, and Shanmuganathan Raman. Eeg2image: Image reconstruction from eeg brain signals. In [ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing \(ICASSP\)](#), pp. 1–5. IEEE, 2023.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Jingyuan Sun, Shaonan Wang, Jiajun Zhang, and Chengqing Zong. Towards sentence-level brain decoding with distributed representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 7047–7054, 2019.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. [arXiv preprint arXiv:2307.09288](https://arxiv.org/abs/2307.09288), 2023.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhajit Som, et al. Image as a foreign language: Beit pretraining for vision and vision-language tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19175–19186, 2023.

Zhenhailong Wang and Heng Ji. Open vocabulary electroencephalography-to-text decoding and zero-shot sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 5350–5358, 2022.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pp. 11328–11339. PMLR, 2020.

Jinzha Zhou, Yiqun Duan, Yu-Cheng Chang, Yu-Kai Wang, and Chin-Teng Lin. Belt:bootstrapping electroencephalography-to-language decoding and zero-shot sentiment classification by natural language supervision. [arXiv preprint arXiv:2309.12056](https://arxiv.org/abs/2309.12056), 2023a.

Jinzha Zhou, Yiqun Duan, Yingying Zou, Yu-Cheng Chang, Yu-Kai Wang, and Chin-Teng Lin. Speech2eeg: Leveraging pretrained speech model for eeg signal recognition. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2023b.

SUPPLEMENTARY MATERIAL FOR BELT-2: BOOTSTRAPPING EEG-TO-LANGUAGE REPRESENTATION ALIGNMENT FOR MULTI-TASK BRAIN DECODING

A RELATED WORKS

EEG decoding Prior brain studies demonstrated the potential to decode speech (Anumanchipalli et al., 2019) and language signals (Anumanchipalli et al., 2019) from the human brain using invasive neuro-sensors, but the risks make it impractical for most people. More recently, a surge of efforts was made to extract rich information from noninvasive brain signals through advanced representation learning techniques, opening the door to a wide array of innovative tasks based on brain signals, such as image reconstruction (Singh et al., 2023) and movement prediction (Zhou et al., 2023b). Nonetheless, Many of these efforts have limitations, including vocabulary size and decoding performance, hindering their suitability for complex practical scenarios. Our work focuses on open-vocabulary sentence decoding from noninvasive brain signals with fluent decoding performance and versatile multi-task adaptability, making it a promising solution for a diverse range of applications.

EEG-Language representation alignment A crucial step for most cross-modality tasks is the acquisition of aligned multi-modal representations (Liu et al., 2023a; Mokady et al., 2021; Rombach et al., 2022). Achieving this involves an alignment step following the acquisition of unimodality pretrained models (Li et al., 2023). Yet, the formidable challenge persists due to the limited scale and sparse EEG dataset annotations, as we strive to create a semantically coherent and universally adaptable EEG encoder, akin to visual counterparts (Dosovitskiy et al., 2020; Radford et al., 2021).

Diverging from the conventional fully-supervised paradigm, infusing natural language supervision enriches non-language modalities representation with semantics and zero-shot generalization (Desai & Johnson, 2021). Previous studies in unimodal vision tasks show that a large vision encoder, trained directly with language supervision, can match performance compared to learning from massive datasets (Joulin et al., 2016). Recent works incorporating language-guided learning also support the value of additional semantics for non-language representation generalization (Wang et al., 2023; Elizalde et al., 2023). Inspired by their successes, our work endeavors to bootstrap the learning of an Encoder that aligns EEG and language representation through natural language supervision.

B MATHEMATICAL SYMBOLS USED IN THIS PAPER

In Table 6 we show a list of mathematical symbols used in this paper.

Table 6: List of mathematical symbols used in this paper

Symbol	Description	Symbol	Description
$\langle \mathcal{E}, \mathcal{S} \rangle$	Word-level EEG embedding sequence and text sentence pair	$\langle \mathcal{E}, c \rangle$	Word-level EEG embedding sequence and sentiment label pair
$\langle \mathcal{E}, \hat{\mathcal{S}} \rangle$	Word-level EEG embedding sequence and text summary pair	$w \in \mathcal{W}$	BPE text token's embeddings
$c \in \mathcal{C}$	Sentiment label	$e \in \mathcal{E}$	EEG embedding vector
		$v \in \mathcal{V}$	Discrete codebook embeddings

C IMPLEMENTATION DETAILS

C.1 IMPLEMENTATION DETAILS FOR THE Q-CONFORMER

The Q-Conformer is implemented using the configuration detailed in Table 7. The detailed structures for the convolution module are shown in Table 8. We use the same Conformer block for the encoder and decoder, each with 2 Conformer blocks. We trained All models are trained on Nvidia A40 GPUs.

Table 7: Detailed configuration of the conformer block

Layer	Hidden Size	Activation Function	Number of Heads
Layer Norm	840	-	-
Feed Forward Module	840	GELU	-
LayerNorm	840	-	-
Multi-Head Self Attention	840	-	8
Convolution	840	-	-
Module	840	-	-
Layer Norm	840	-	-
Feed Forward Module	840	GELU	-
LayerNorm	840	-	-

Table 8: Detailed configuration of the convolution module

Layer	Kernal	Stride	In Channel	Out Channel
Layer Norm	-	-	840	840
Pointwise Convolution	1	1	840	2×840
Depthwise Convolution	31	1	840	840
Batch Norm	-	-	840	840
Pointwise Convolution	1	1	840	840
Dropout	-	-	-	-

C.2 TRAINING DETAILS FOR EEG-TO-LANGUAGE ALIGNMENT LEARNING

To train the Q-Conformer during the EEG-to-language alignment learning, we use a weighted summation of all the following loss terms:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{vq} + \lambda_2 \mathcal{L}_{bpe} + \lambda_4 \mathcal{L}_{elm} + \lambda_3 \mathcal{L}_{neg}, \quad (8)$$

λ_1 to λ_4 are coefficients for each loss term. We set λ_1 to λ_4 as [1, 10, 10, 0.001]. The main reason for such a setting is the aim to prioritize the learning of achieving EEG-to-language alignment and the training of the query prompt specific to the ELM task. To avoid collapse in training, we implemented the gradient normalization method to normalize the scale of the loss function and stabilize the training process.

C.3 TRAINING VIRTUAL PREFIX FOR BRIDGING Q-CONFORMER AND LLM

The prefix-tuning method used in our paper closely follows the implementation in Li & Liang (2021), the objective function (\mathcal{L}_{bridge}) is defined as a modified loss function tailored to guide the selective of continuous virtual prefix prompts. We use θ to denote the matrix that stores the virtual prefix. Using the machine translation loss \mathcal{L}_{tr} as an example, the objective function can be expressed as:

$$\mathcal{L}(\theta_{\text{bridge}}) = \mathcal{L}_{tr}(\hat{\mathcal{S}}, \mathcal{S}) \quad (9)$$

In this example, the prefix prompts to learn properly describe the EEG-to-Language translation task to the subsequence frozen LLM, utilizing the generation capacity of the LLM models to improve translation performance.

C.4 TRAINING DETAILS FOR MULTI-TASK LEARNING

To extend our model to multi-task decoding, we simultaneously train the model in three EEG decoding tasks including translation, summary, and sentiment classification task. We randomly sample a task for each batch during the training epochs. The loss function for translation task \mathcal{L}_{tr} and sentiment classification tasks \mathcal{L}_{st} are illustrated in Equation 6 and Equation 7 respectively.

For learning the summary task, the loss function could be written as follows:

$$\mathcal{L}_{sum} = - \sum_l^{|\hat{\mathcal{S}}|} \log p(s_l \in \hat{\mathcal{S}}) \quad (10)$$

, where $p(s_l)$ denotes a model predicting the word token for the next location. The final multi-task objective \mathcal{L} is written as follows:

$$\mathcal{L}_{mt} = \mathcal{L}_{tr} + \mathcal{L}_{sum} + \mathcal{L}_{st} \quad (11)$$

D IMPROVED Q-CONFORMER EEG ENCODER

We observed a noteworthy trend when utilizing a relatively larger learning rate of $1e - 4$, as opposed to the optimal learning rate of $5e - 6$ for the top-performing Q-Conformer Encoder, as indicated in Figure 8. This variance in learning rates led to a remarkable performance by the Q-Conformer Encoder on the training dataset, resulting in notably high BLEU Scores. Specifically, the BLEU-1 and BLEU-4 scores soared to remarkable levels, reaching 93.03 and 92.69 respectively. In stark contrast, the EEG-to-Text baseline method significantly lagged behind, registering only BLEU-1, 4 scores of 38.98 and 6.82 during our replicated training, highlighting the superior EEG encoding capabilities of the Q-Conformer Encoder.

It's also worth noting that the BLEU-1 performance of the Q-Conformer encoder experienced a decline from 42.43 to 35.48 during the testing phase, we interpret this as a minor setback. Such a reduction in performance can often be attributed to the challenges of generalization, which frequently happen in the context of training on a relatively small dataset.

Furthermore, it's worth highlighting that within this setting, the Q-Conformer still achieved a testing BLEU-4 score of 9.3, surpassing the baseline EEG-to-Text method's training set BLEU-4 score. This outcome serves as a compelling testament to the enhanced encoding capacity conferred by our Q-Conformer Encoder.

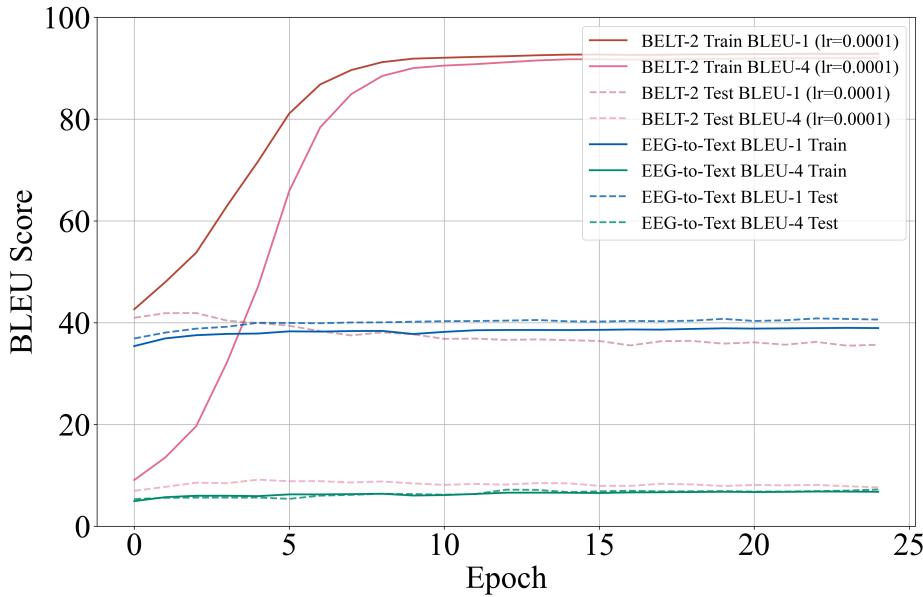


Figure 8: EEG encoder performance comparison

E COMPARISON WITH BELT-2 WITHOUT BPE-LEVEL CONTRASTIVE LEARNING

In Figure 9(a) and Figure 9(b), we present a comprehensive comparison of the learning curves and BLEU-1 curve of the baseline EEG-to-Text model (Cruttenden, 2014), the Q-Conformer encoder without applying the BPE-level contrastive learning (BELT-2 w/o BPE-CT) and the Q-Conformer

encoder with BPE-level contrastive learning (BELT-2 w/ BPE-CT)g. The visualized learning curves include the BLEU-1 score and loss values for 30 epochs on the test split. Comparing the EEG-to-Text model and the BELT-2 model, it’s evident that BELT-2 offers a significant reduction in loss values with or without BPE-level contrastive learning, indicating the proposed model architecture is more efficient in capturing EEG patterns. However, a notable observation arises after epoch 8. Without the BPE-contrastive learning (orange curves), the BLEU-1 score fluctuates and drops significantly. On the contrary, the introduction of BPE-level loss helps stabilize the model’s performance, particularly on unseen EEG data. This highlights the substantial enhancement brought about by our proposed BPE-contrastive learning framework.

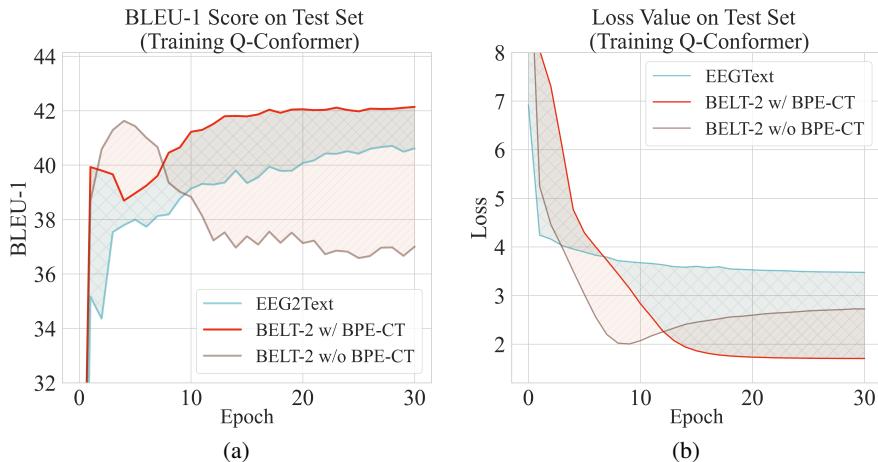


Figure 9: Ablation Study on Different Settings

F MULTI-TASK TRAINING RESULTS

We show the performance of translation, summary, and sentiment classification on the test set during the multitask training learning phase of BELT-2 in Table 10. In Table 10(a), we can observe that without the use of pretrained weights, all tasks are learned from scratch. In this case, the translation BLEU-1 score starts from 4.06 BLEU-1 score and rises to only reaches 41.47 and the summarization BLEU-1 score reaches 28.72. Also, the sentiment classification accuracy gradually increased to 59%. However, the use of Q-Conformer pretrained on translation tasks could improve the training stability and performance of both the sentiment classification task and the summarization task. Due to the pretrained weights, we observed that in Table 10(b), the BLEU-1 score of the summarization performance and sentiment achieved 23.0 BLEU-1 score after the first training epoch. Then continued to increase to 31.17. The accuracy for sentiment classification also reaches 79.86% at its peak and stabilizes at around 74%. However, the performance of the translation task slightly decreased. This is an expected phenomenon in multi-task training. Nonetheless, this ethernet still shows the multi-task learning capacity and extensibility of our BELT-2 framework.

G GENERATED SUMMARIZATION RESULTS

We created the summarization dataset with the prompt "Rewrite the sentence by summarizing its main idea using 8 words from the sentence and keep the summarized sentence similar to the original sentence: $\{s\}$ " where $\{s\}$ is the original sentence from the dataset. Table 9 showcases summary and prediction samples generated by the BELT-2 model. We could see those summary ground truths cover the key ideas of the original sentence and are within the maximum summarization word limit. On the training set, our BELT-2 model could learn and precisely generate a summary of the EEG signal, such as "film with twists" vs. "film with twists.". However, this summarization capacity did not generalize well on unseen test and validation data. We consider the lack of training data as one of the major reasons for this problem. Another reason is that our current model lacks higher-level

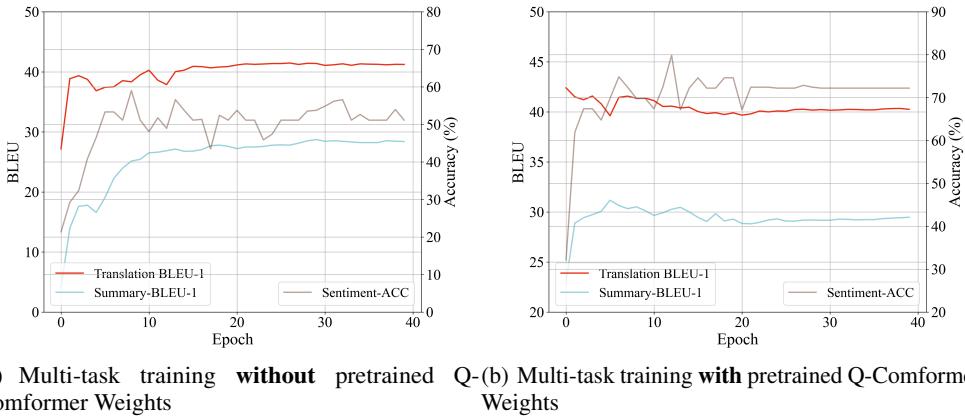


Figure 10: Ablation study on multitask learning and effect of our pretrained weights

skill that requires additional reasoning and abstraction skills beyond the mere translation of the brain signal, which leaves room for future improvements.

H ABLATION EXPERIMENTS ON HYPER-PARAMETERS

We conducted an ablation study on different hyper-parameters including the learning rate, batch size, frequency of the inserted cross-attention layer in the context layer of the Q-Conformer, and the number of querying prompts. The evaluation metrics can be found in Figure 11. We observe that the introduction of BPE-contrastive learning consistently improves training stability and model performance in different hyper-parameter settings. This result shows that the learning performance of BELT-2’s EEG encoder is not easily affected by the change of training parameters and is relatively easy to reproduce.

I AUGMENTATION EFFECT OF SPECULATIVE AUGMENTATION

The limitation of unique sentence from the training dataset also limits the diversity of the MLC context outputted by the Q-Conformer. The training set we used in our cross-sentence setting contains only 790 unique sentences as target for prefix-tuning when bridging Q-Conformer and LLM. For the Q-Conformer, predicts around 900 uniques MLC throughout the training dataset. This lack of training inputs makes the training for a good virtual prefix difficult. To solve this problem, our speculative augmentation method reuse cached MLC from the training stage of Q-Coformer. When using MLC from $K = 15$ checkpoints, we achieve a total of 5107 samples for prefix-tuning.

J EXTENSIVE EXAMPLES OF GENERATED TRANSLATION OUTPUTS

We provide extensive translation outputs from our BELT-2 model compared with the baseline EEG-to-Text model and the ground truth in Table 10. It shows that for some samples, the BELT-2 model still has insufficient performance, which indicates room for future improvements.

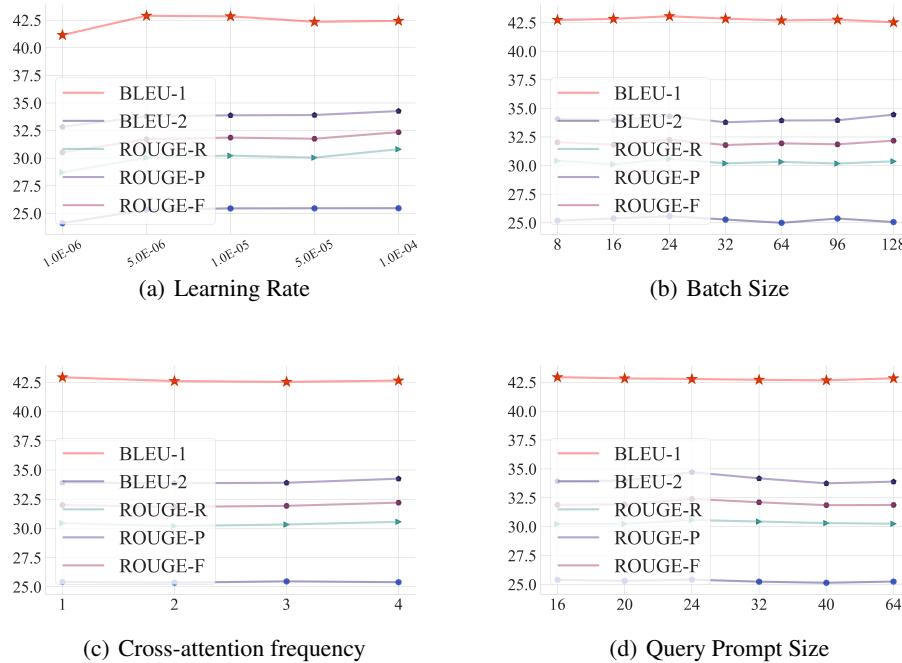


Figure 11: Ablation study on hyper-parameters.

Table 9: Summarization examples and generated results on the train set. The **bold** denotes an exact match between the ground truth and our prediction. underline denotes a fuzzy match with similar semantic meanings.

Training		
(1)	Sentence	Beautifully crafted, engaging filmmaking that should attract upscale audiences hungry for quality and a nostalgic, twisty yarn that will keep them guessing.
	Summary GT	High-quality film with twists.
	Prediction	-quality film with twists.
(2)	Sentence	Slow, silly and unintentionally hilarious.
	Summary GT	Silly, slow comedy .
	Prediction	inger, slow movie .
(3)	Sentence	The movie is for fans who can't stop loving anime, and the fanatical excess built into it.
	Summary GT	Anime fans will love excessive movie.
	Prediction	imated fans will love this gore.
(4)	Sentence	But here's the real damn: It isn't funny, either.
	Summary GT	Funny, but not really .
	Prediction	unny, smart not really .
(5)	Sentence	Everything was as superficial as the forced New Jersey lowbrow accent Uma had.
	Summary GT	Uma's accent was fake .
	Prediction	ma's accent was fake .
(6)	Sentence	Feels like nothing quite so much as a middle-aged moviemaker's attempt to surround himself with beautiful, half-naked women.
	Summary GT	Filmmaker surrounds himself with beautiful women .
	Prediction	mmakers imagined himself with beautiful women .
(7)	Sentence	He died in Springport, New York in 1815.
	Summary GT	Man passed away in Springport .
	Prediction	passed away in Springport .
Test and Validation		
(1)	Sentence	A richly imagined and admirably mature work from a gifted director who definitely has something on his mind.
	Summary GT	Director's mature work reflects deep thoughts.
	Prediction	's debut film. his empathy.
(2)	Sentence	An amateurish, quasi-improvised acting exercise shot on ugly digital video.
	Summary GT	Ugly video showcases poor acting .
	Prediction	ma., ugly acting .
(3)	Sentence	Warm Water Under a Red Bridge is a quirky and poignant Japanese film that explores the fascinating connections between women, water, nature, and sexuality.
	Summary GT	Japanese film explores women, water, nature, sexuality poignantly.
	Prediction	actor, themes's love, and. love.eticsancy.
(4)	Sentence	It just doesn't have much else... especially in a moral sense.
	Summary GT	Limited moral compass
	Prediction	role compass .
(5)	Sentence	It's solid and affecting and exactly as thought-provoking as it should be.
	Summary GT	Thought- provoking and solid.
	Prediction	inful provoking film funny.
(6)	Sentence	The art direction is often exquisite, and the anthropomorphic animal characters are beautifully realized through clever makeup design, leaving one to hope that the eventual DVD release will offer subtitles and the original Italian-language soundtrack.
	Summary GT	Beautiful animal characters, DVD subtitles.
	Prediction	iful, inter. funny experience.

Table 10: Extensive examples of generated translation outputs from unseen EEG signals in the test set. The **bold** denotes an exact match while underline denotes a fuzzy match with similar semantic meanings.

(1)	Target	It's not a particularly good film, but neither is it a monsterous one .
	Others	was a a bad good story, but it is it bad bad. one.
	Ours	It's not a bad bad movie, but it is it kinda good bad one .
(2)	Target	It's solid and affecting and exactly as thought- provoking as it should be .
	Others	was a, it, it what it.provoking as it is be.
	Ours	It's , believable, is what - provoking as the sounds be .
(3)	Target	Co-writer/director Jonathan Parker's attempts to fashion a Brazil-like, hyper-real satire fall dreadfully short.
	Others	operfounder of director of Dem is novel to make a film-themed film but-realistic of flatfully short of
	Ours	Theenstarrings director John Dem hass films to make a new-style film -realisticromre are flatareadfully flat.
(4)	Target	After World War II , Kennedy entered politics (partly to fill the void of his popular brother, Joseph P. Kennedy , Jr., on whom his family had pinned many of their hopes but who was killed in the war).
	Others	the War II, the was the andasly as serve the void left a father father , John Kennedy. Kennedy , who.) who the he father had been their of his hopes). never was never in the war).
	Ours	After the War II , became politics,andy to fulfill the void left his father father, John Kennedy . Kennedy, who.,who the Kennedy family had placedbased their of their hopes). had had in Battle.
(5)	Target	It's solid and affecting and exactly as thought- provoking as it should be.
	Others	was a, it, it what it.outoking as the sounds be.
	Ours	It's , logical, is what - provoking as the sounds be.
(6)	Target	Too much of this well-acted but dangerously slow thriller feels like a preamble to a bigger, more complicated story, one that never materializes.
	Others	bad of a is-known, not over- is like a film-ble to a more, more dramatic story. which that will quiteizes.
	Ours	Too much drama is-made, unly un-. like a -ble to a much, more serious,. one that' quiteizes.
(7)	Target	In 1923 he was awarded the inaugural Bôcher Memorial Prize by the American Mathematical Society.
	Others	the, married born the Nobel Pulitzercentne Prize Medal for the French Academyical Society.
	Ours	In 1815,he was awarded the Pulécher Prize Prize , the Royal Academyematical Society.
(8)	Target	He later became an educator, teaching music theory at the University of the District of Columbia; he was also director of the District of Columbia Music Center jazz workshop band.
	Others	was became a actor and and at and the University of California Arts of Columbia. and also also a of the University of Columbia 's School. department..
	Ours	He later became associate at and at at at the University of California West of Columbia and and he also of the English' Columbia ' Department. department.
(9)	Target	Fans of the TV series will be disappointed, and everyone else will be slightly bored.
	Others	of the film show " remember familiar to however the will will be happy amused.
	Ours	Fans of the movie series will be , as the who will be left disappointed.