HIERARCHICAL LATENT FACTORIZATION FOR NON-INVASIVE SEQUENTIAL MOTOR DECODING

Anonymous authors
Paper under double-blind review

ABSTRACT

Non-invasive neural decoding is challenged by context-dependent representation drift: during sequential movements, the same finger movement can produce different neural patterns across participants, sequences, and serial positions. These sources of variation operate over different temporal scales, and mix nonlinearly, hindering stable decoding. We formulate this setting as primary-task prediction from mixed, context-dependent observations and propose HiLa, a **hi**erarchically **la**tent factorization framework for stable decoding during sequential movements. HiLa uses hierarchical labels as supervisory signals and conditional branch splitting to learn context-robust representations that preserve action identity across contexts. HiLa achieves 62.02% movement identity accuracy during sequential finger tapping and 50.44% keystroke accuracy during typing. HiLa's backbone outperforms benchmarks, while hierarchical supervision improves accuracy, with gains remaining stable in cross-participant and cross-context tests. Latent factorization yields more decontextualized and biologically plausible representations without sacrificing decoding accuracy. Together, these results show that hierarchically supervised latent factorization can improve primary-task decoding under structured contextual variation in non-invasive neural recordings.

1 INTRODUCTION

Compared with invasive recording, non-invasive recording is the more practical route to decoding naturalistic movement at scale: it requires no surgery, reaches the general population, and supports goals beyond control signals (Ding et al., 2026; Edelman et al., 2025). Magnetoencephalography (MEG) non-invasively measures magnetic fields generated by neuronal activity, providing higher signal-to-noise ratios than electroencephalogram (EEG) for superficial cortical sources, such as the sensorimotor cortex(Piastra et al., 2020; Goldenholz et al., 2009; Lopes da Silva, 2013), and millisecond-scale temporal resolution compared with functional magnetic resonance imaging (fMRI) (Logothetis, 2008; Hall et al., 2014). Recent advances in wearable MEG enable recordings outside conventional shielded rooms, further expanding the potential of MEG for continuous, naturalistic movement decoding and BCI applications (Brookes et al., 2022; Boto et al., 2018).

Movement decoding from non-invasive neural recordings is a supervised prediction problem over spatiotemporal neural dynamics, complicated by mixed observations and contextual distribution shifts (Hämäläinen et al., 1993; Gross, 2019). First, sensors capture distributed whole-brain activity, combining movement-related signals with activity associated with other cognitive processes, such as vision, attention, planning, and sensory feedback (Baillet, 2017; Hari et al., 2010). Second, the input–target relationship is intrinsically non-stationary: the neural activity underlying the same movement varies across behavioral contexts and evolves over time within an individual (Rule et al., 2019; Driscoll et al., 2022), a phenomenon commonly referred to as representational drift (Churchland et al., 2012; Russo et al., 2018; Shenoy & Kao, 2021). These factors overlap in time across sensors, leading to temporally unstable representations of individual movements (Lévy et al., 2026). The key challenge is therefore to learn movement representations that are robust across contexts while leveraging structured behavioral context.

Sequential movement provides a concrete setting for this challenge. We studied two tasks: a memorized finger-sequence task (Kornysheva et al., 2019), in which participants execute learned sequences by pressing their five fingers in a prescribed order, and a sentence-typing task (Lévy

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

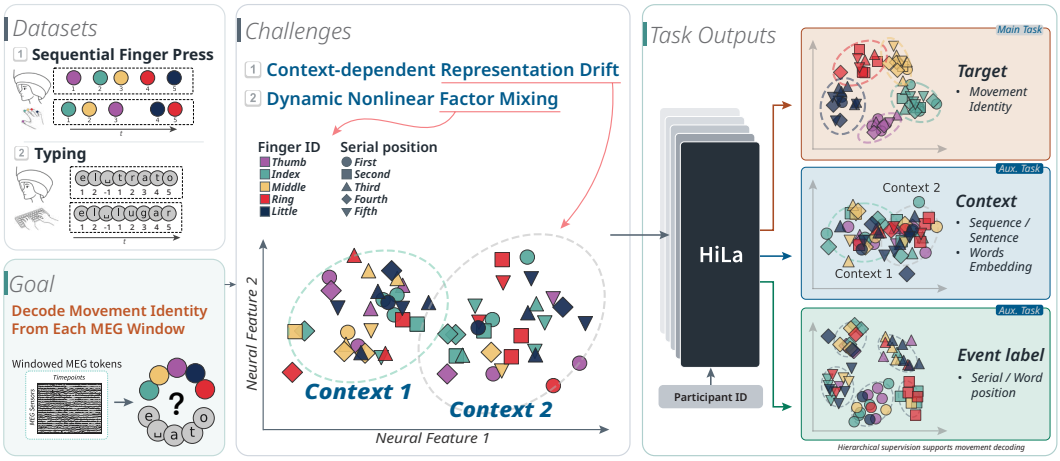

Figure 1: Overview of HiLa.

et al., 2026), in which participants produce Spanish sentences through keystrokes. In both tasks, a sequence or sentence comprises an ordered series of discrete events. Each event is defined by its action identity (finger or key) and serial position within the sequence, while the sequence or sentence defines the event's higher-level context. This organization is consistent with evidence that motor networks represent both individual movement identities and sequence-level structure, including where each movement occurs in the sequence and how consecutive movements are temporally organized (Yokoi & Diedrichsen, 2019; Averbeck et al., 2002; Kornysheva & Diedrichsen, 2014). We exploit this naturally occurring hierarchy to guide movement decoding.

To address this challenge, we develop HiLa, a hierarchical latent representation framework that uses behavioral labels at multiple levels to guide representation learning. HiLa first establishes a strong representation for individual movement events, outperforming top motor imagery models. Hierarchical supervision further improves individual-event decoding, with gains generalizing to held-out subjects and contexts. Conditional branch splitting reduces correlations between latent representations while preserving decoding performance, whereas adversarial losses improve auxiliary-label prediction only marginally without improving the primary task. Finally, we verify that HiLa learns biologically plausible representations consistent with established knowledge of sequential movement and relies on expected sensorimotor sources, providing evidence that the learned representations capture meaningful movement-related structure.

The main contributions are as follows: (1) We introduce HiLa, a branch-structured architecture for learning latent representations from MEG and decoding individual movement events. (2) We formulate sequential motor decoding as a mixed-source problem and show that jointly supervising movement and contextual variables improves movement decoding and generalizes across participants and sequences. (3) We show that movement- and context-related information is partially specialized yet remains coupled in MEG representations and enforcing stronger invariance or decorrelation do not induce better latent.

2 Related Work

HiLa's encoder backbone is a compact convolutional architecture based on the established temporal–spatial factorization of EEG decoders (Schirrmeister et al., 2017; Lawhern et al., 2018). Along the spatial axis, it retains sensor-wise spatial filtering rather than incorporating explicit sensor geometry or connectivity structure (Lévy et al., 2025; Klepl et al., 2024; Li et al., 2021). Along the temporal axis, HiLa uses multi-scale temporal convolutions (Zhang et al., 2021; Song et al., 2023; Altaheri et al., 2022; 2025). The convolutional backbone is followed by Transformer blocks with axial attention, alternating between temporal and channel-wise attention (Ho et al., 2020).

On top of this backbone, HiLa addresses the mixed-source nature of neural recordings. In invasive settings, task-relevant dynamics can be separated from other sources of variability by exploiting high-dimensional population activity through linear projections or factorization. For example, demixed PCA separates task-related and temporal dimensions through linear projections (Mante et al., 2013;

Kobak et al., 2016), while tensor methods extend this factorization to neuron × time × trial data, disentangling within-trial dynamics from across-trial variability and other shared structure (Williams et al., 2018; Pellegrino et al., 2024). These methods typically rely on individually resolved neurons and repeated, time-locked trials. In contrast, continuous non-invasive recordings provide sensor-level signals in which each sensor reflects a collinear mixture of neural sources (Schoffelen & Gross, 2009), motivating a different approach in HiLa.

A complementary line of work formulates invasive neural decoding as a sequence-to-sequence (seq2seq) problem, modeling neural activity as evolving latent trajectories. These methods infer trial-specific trajectories (Yu et al., 2009; Pandarinath et al., 2018), identify behaviorally relevant or input-driven latent dimensions (Sani et al., 2021; Vahidi et al., 2024), and learn nonlinear or behavior-conditioned dynamics (Hurwitz et al., 2021; Vahidi et al., 2025; Schneider et al., 2023; Gosztolai et al., 2025). Although appealing for non-invasive decoding, seq2seq modeling is challenging for noisy MEG in online settings, where individual timesteps contain limited information. Most non-invasive decoders therefore aggregate neural activity within temporal windows.

In non-invasive neural decoding, major efforts address participant- and session-level variability to improve cross-recording generalization. One strategy aligns data before a shared decoder, either statistically by re-centering, re-scaling, or rotating participant-specific sensor covariances (He & Wu, 2020; Zanini et al., 2018; Mellot et al., 2023), or through learned participant-specific input stages (Défossez et al., 2023; Scotti et al., 2024). Recent work shows that task-supervised subject-specific encoders can largely reproduce the effect of explicit covariance alignment (Lopes et al., 2026). HiLa adopts this strategy using a participant-specific two-layer MLP over the channel dimension to absorb subject-dependent spatial variation.

A second strategy seeks participant-invariant representations by removing subject-specific variation. Domain-adversarial training has been applied either to the full representation (Özdenizci et al., 2020; Rayatdoost et al., 2021; Bethge et al., 2022) or to separate task and nuisance branches (Han et al., 2020; Jing et al., 2025), while contrastive methods align representations across participants (Shen et al., 2023). Related disentanglement methods encourage latent factors to be decorrelated or statistically independent (Cogswell et al., 2015; Higgins et al., 2017; Kim & Mnih, 2018; Chen et al., 2018), although unsupervised disentanglement generally requires additional inductive biases for identifiability (Locatello et al., 2019). HiLa takes a complementary approach: rather than removing behavioral variation, it uses behavioral variables to organize that variation into separate latent branches. We find that adding adversarial or decorrelation objectives to the full representation or individual branches does not further improve primary-task decoding.

This use of behavioral variables as supervision connects HiLa to multi-task representation learning, where auxiliary targets provide complementary supervision for a shared representation (Ruder, 2017; Zhang & Yang, 2021), and to hierarchical learning, where representations preserve information across levels of a label hierarchy (Søgaard & Goldberg, 2016; Sanh et al., 2019; Zhang et al., 2022). In neural decoding, however, higher-level context has primarily been used at the output level, for example through language models that constrain decoded sequences (Willett et al., 2023; Metzger et al., 2023; Card et al., 2024; Lévy et al., 2025). HiLa instead uses sequence or sentence context and event position as auxiliary targets to structure the neural representation itself. This hierarchical supervision improves primary-task decoding and generalizes across held-out participants and behavioral contexts.

3 METHODS

3.1 PROBLEM FORMULATION

We formulate sequential motor decoding as primary-task prediction from mixed, context-dependent observations. Let $W_n^{(r)} \in \mathbb{R}^{T \times C}$ denote an MEG window from recording session r, with T time points and C sensors. The primary task is to predict its movement target $y_n^{(r)} \in \mathcal{Y}$, along with contextual and event-related labels $\mathbf{c}_n^{(r)} \in \mathcal{C}, \mathbf{e}_n^{(r)} \in \mathcal{E}$. We model the observed activity as a nonlinear mixture of movement-, context-, and event-related factors: $W_n^{(r)} = g\left(z_n^{(a)}, z_n^{(c)}, z_n^{(e)}\right) + \epsilon_n^{(r)}$. Our goal is to learn a representation that supports robust movement decoding while organizing contextual and event-related information into structured latent factors.

3.2 Model Overview

The model is trained end-to-end with two components (Fig. 2): a shared MEG encoder and a factorized latent splitter. The encoder transforms each MEG window through participant-specific alignment, sensor mixing, multiscale temporal convolution, and factorized spatial–temporal attention. The latent splitter parameterizes and factorizes the representation into sequence, order, and finger branches, and applies branch-specific temporal aggregation followed by linear prediction heads.

Participant Adapter. To account for participant-specific channel mixing (Gross et al., 2013), we

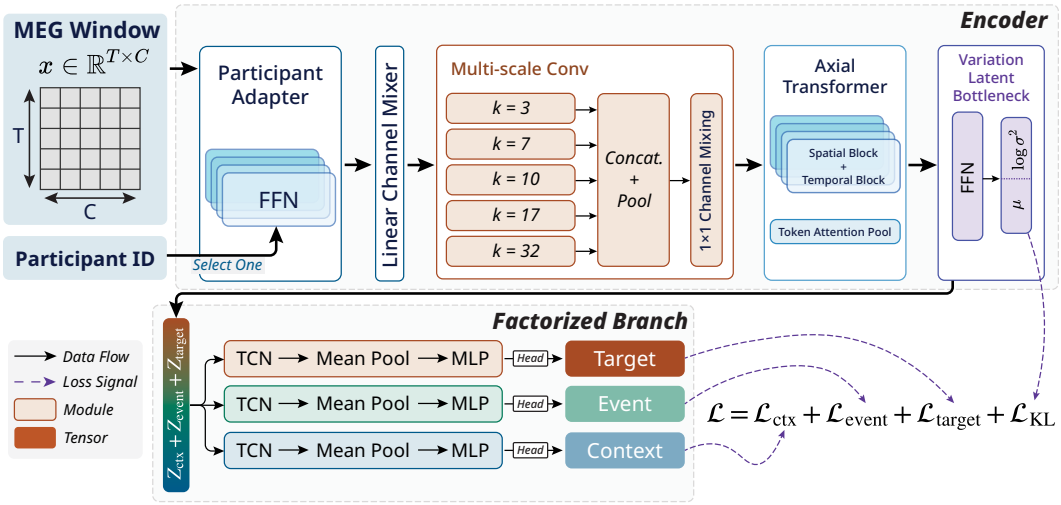

Figure 2: Architecture of HiLa.

apply a participant-specific two-layer MLP along the channel dimension (Ramoser et al., 2000): $\widetilde{W}_n^{(r)} = A_{s^{(r)}}\left(W_n^{(r)}\right), A_s : \mathbb{R}^S \to \mathbb{R}^C$, where A_s maps C channels through a hidden layer and back to C channels, preserving $W_n^{(r)}, \widetilde{W}_n^{(r)} \in \mathbb{R}^{T \times C}$.

Linear channel mixer. A shared linear transformation is applied along the channel dimension: $H_n^{(r)} = \widetilde{W}_n^{(r)} W_{\text{proj}}, W_{\text{proj}} \in \mathbb{R}^{S \times d_{\text{out}}}$, mapping the C sensor channels to d_{out} features while preserving the temporal dimension, so that $H_n^{(r)} \in \mathbb{R}^{T \times d_{\text{out}}}$.

Multiscale temporal convolution. To capture temporal structure across canonical neural frequency ranges, five parallel temporal convolution branches are used, with kernel sizes loosely sampling temporal dynamics from gamma, beta, alpha, theta, and delta ranges. Each branch produces K temporal feature maps. Concatenating the outputs along the feature axis yields $X_n^{\text{mtc}} \in \mathbb{R}^{T \times (FK) \times d_{\text{out}}}$.Temporal pooling with kernel size 2 and stride 2 then reduces the temporal dimension to $T/2$: $\tilde{X}_i^{\text{mtc}} \in \mathbb{R}^{\frac{T}{2} \times (F \cdot K) \times d_{\text{out}}}$. The pooled features are then passed to the subsequent token-level encoder.

Pointwise channel-mixing transformation At each pooled time point, the $F \cdot K$ features are flattened together with their d_{out}-dimensional representations: $\widetilde{x}_{i,t} \in \mathbb{R}^{FKd_{\text{out}}}$. A shared pointwise channel-mixing transformation is then applied independently at each time point: $\widehat{x}_{i,t} = \text{SiLU}\left(\text{BN}\left(W_{\text{mix}}\widetilde{x}_{i,t}\right)\right), W_{\text{mix}} \in \mathbb{R}^{FKd_{\text{out}} \times FLd_{\text{out}}}$. Implemented as a 1×1 convolution, this transformation mixes the flattened band, latent, and feature dimensions without temporal mixing. The resulting features are reshaped into $P = FL$ tokens: $M_{i,t} \in \mathbb{R}^{P \times d_{\text{out}}}, P = FK$.

Spatial–temporal Transformer. The token representations are then projected to the Transformer dimension D and augmented with learned token and temporal embeddings, yielding $X \in \mathbb{R}^{B \times T/2 \times P \times D}$. An axial spatiotemporal Transformer separately models interactions across tokens and time. Each Transformer block alternates spatial self-attention across the P tokens at each time point with temporal self-attention across the $T/2$ time points for each token. After N blocks and layer normalization, $Z = \text{LN}(X^{(N)}) \in \mathbb{R}^{B \times T/2 \times P \times D}$. The resulting features are aggregated by learned softmax pooling over tokens at each time point before the variational bottleneck and task-specific prediction heads.

Latent Factorization and Prediction Heads. From the shared encoder representation, we parameterize a diagonal Gaussian variational posterior, $(\mu_i, \log \sigma_i^2) = g_\phi(X_i^{\text{att}}), q_\phi(z_i \mid x_i, s_i) = \mathcal{N}\big(\mu_i, \text{diag}(\sigma_i^2)\big)$. We use the posterior mean μ_i as the latent representation for the subsequent branch encoders and task heads, without sampling from the posterior. The posterior is regularized toward a standard normal prior using $\mathcal{L}_{\text{KL}} = D_{\text{KL}}(q_\phi(z_i \mid x_i, s_i) \,\|\, \mathcal{N}(0, I))$.

The posterior mean is partitioned into three branch-specific latent sequences corresponding to context, event, and target information: $\mu_i = [\mu_i^{\text{context}}, \mu_i^{\text{event}}, \mu_i^{\text{target}}]$, where $\mu_i^b \in \mathbb{R}^{T' \times D_b}$. Each branch is independently encoded along the temporal dimension and pooled to obtain a fixed-dimensional branch representation, $h_i^b = W_b \,\text{Pool}_t(\text{TCN}_b(\mu_i^b)) + b_b$, where $h_i^b \in \mathbb{R}^{d_{\text{cond}}}$. The context and event branches predict their corresponding auxiliary labels, while the target branch predicts the primary target: $\hat{\mathbf{c}}_i = f_{\text{context}}(h_i^{\text{context}})$, $\hat{\mathbf{e}}_i = f_{\text{event}}(h_i^{\text{event}})$, and $\hat{y}_i = f_{\text{target}}(h_i^{\text{target}})$. The model is trained jointly with prediction and variational objectives:$\mathcal{L} = w_{\text{context}}\mathcal{L}_{\text{context}} + w_{\text{event}}\mathcal{L}_{\text{event}} + w_{\text{target}}\mathcal{L}_{\text{target}} + w_{\text{KL}}\mathcal{L}_{\text{KL}}$. The prediction terms supervise the corresponding label branches and $\mathcal{L}_{\text{KL}}$ regularizes the variational bottleneck.

4 Dataset and training details

Neuron2019. The dataset comprises a sequential finger-tapping task in which each participant learned four randomly generated five-finger sequences, arranged in a 2×2 factorial design crossing finger order and timing pattern. Continuous MEG was sampled at 100 Hz from 273 sensor channels. Trials were segmented into 500-ms windows with a 40-ms stride.

For each press, the EMG-defined onset $\tau_{\text{onset}}^{(r,p)}$ was used to define a 500-ms press interval, $\mathcal{P}_p^{(r)} = [\tau_{\text{onset}}^{(r,p)} - 400\,\text{ms}, \tau_{\text{onset}}^{(r,p)} + 100\,\text{ms})$. For each MEG window, the finger and event labels were assigned according to the press interval with the largest relative overlap with the window, provided that the overlap was at least 80%; otherwise, the window was labeled as non-press (0). The target label ($y \in \{0, \ldots, 5\}$) denotes the pressed finger, the event label ($e \in \{0, \ldots, 5\}$) denotes its serial position, and the context label ($c \in \{0, \ldots, 55\}$) denotes the sequence identity. Cross-entropy loss was applied to all three labels. Data were split at the trial level within each participant and sequence using a 5-fold cross validation and 4 seeds; all windows inherited the assignment of their parent trial, preventing leakage across splits.

SpanishBCBL (Brain2Qwerty). The dataset comprises a sentence-typing task in which participants typed memorized Spanish sentences on a QWERTY keyboard following an RSVP cue. We used the Brain2Qwerty V1 MEG cohort, comprising 19 participants and 306 MEG channels, with recordings band-pass filtered at 0.1–20 Hz and down-sampled to 50 Hz. Each key press defined a 500-ms analysis window spanning -200 to $+300$ ms relative to the key onset.

Each window was assigned three labels: the target label ($y \in \mathcal{Y}$, $|\mathcal{Y}| = 27$) denotes the pressed key, including space; the event label ($e \in \mathcal{E}$, $|\mathcal{E}| = 40$, 40 to accommodate typo) denotes the character position within the word, space is dropped from this loss, and the *context* labels comprise word and sentence identity. The word context was supervised using an MSE loss against the corresponding LLM embedding, while sentence was supervised using cross-entropy loss.

Splits followed the original dataset protocol. Sentences were grouped by TF–IDF cosine similarity (threshold 0.5), and groups were assigned entirely to train, validation, or test to prevent similar sentences from crossing splits. The resulting split targeted an 80/10/10 split of keystrokes, with each window inheriting its sentence's partition. Validation was used for model selection and test for final evaluation, with 20 seeds per condition.

5 Experiments

5.1 Benchmark with Adapted EEG/MEG Baselines

We first compare the HiLa encoder backbone with established non-invasive neural decoding architectures: Linear; EEGNet (Lawhern et al., 2018), a widely used lightweight EEG baseline; ATCNet (Altaheri et al., 2022) and TCFormer (Altaheri et al., 2025), strong performers in motor-

imagery decoding; and Brain2Qwerty (Lévy et al., 2026), the convolution encoder architecture used in the MEG-to-text decoding paper, without the sentence transformer.

All models were adapted to the same movement-target-only decoding task on both the Neuron2019 (sequential finger-tapping) and SpanishBCBL (typing) datasets and evaluated using identical procedures. HiLa achieved 60.42% finger accuracy on the sequential finger-tapping task, compared with 55.50% for Brain2Qwerty, while using substantially fewer parameters (5.47M vs. 237M). On the typing dataset, HiLa similarly achieved higher character accuracy than the baseline. The full model achieved higher accuracy with fewer parameters because of the smaller branch configuration.

Model	Neuron 2019 Finger Acc.	Neuron 2019 Size	SpanishBCBL Char. Acc.	SpanishBCBL Size
Linear	34.95 ± 0.19	1.6K	15.15 ± 0.02	8.3K
EEGNet	47.78 ± 0.42	120K	33.14 ± 0.36	1.14M
TCFormer	51.13 ± 0.27	3.36M	35.71 ± 1.38	119M
ATCNet	52.60 ± 0.13	1.81M	40.17 ± 0.13	65.5M
Brain2Qwerty	55.50 ± 0.17	237M	31.61 ± 0.87	238M
HiLa (target only)	60.42 ± 0.57	5.47M	49.60 ± 1.07	29.7M
HiLa (full)	$\mathbf{62.02 \pm 0.70}$	5.37M	$\mathbf{50.44 \pm 0.90}$	22.8M

Table 1: Target movement decoding performance and model size of EEG/MEG baselines (mean ± SD, %).

5.2 Ablation

We next evaluated the contribution of individual components of HiLa by measuring the change in finger-decoding accuracy relative to the full model (Fig. 3). Removing the participant adapter produced the largest performance reduction, highlighting the importance of participant-specific adaptation. Removing the multiscale temporal convolution also substantially reduced accuracy, while replacing it with a single-scale convolution resulted in a smaller but significant reduction. Removing the pointwise convolution, branch TCN, or axial attention, or removing the variational bottleneck produced more modest but significant decreases in accuracy.

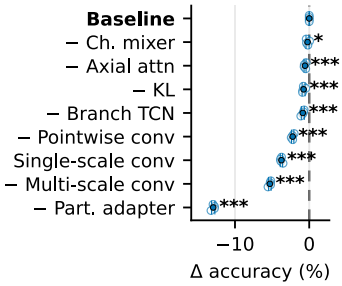

Figure 3: Ablation (mean±SD). Pairwise Wilcoxon signed-rank tests with BH-FDR correction.

5.3 Hierarchical Label Supervision Improves Decoding

We first tested whether contextual and event-level supervision improves target decoding. To isolate the effect of supervision, all models used the same architecture and differed only in their supervised targets and loss terms (Table 2). Across both datasets, serial-position supervision produced larger gains in primary-task decoding than sequence or sentence-context supervision, while joint supervision of all three targets consistently achieved the highest target accuracy. Word- and sentence-level labels were more difficult to decode and provided only marginal gains. Sentence accuracy was not reported because the test split contained unseen sentences. Using separate branches for the three targets did not significantly change primary-task performance relative to the single-branch design. Additional accuracy breakdowns by individual finger/key and participant are reported in the supplementary tables.

Table 2: Decoding performance under different supervision (mean ± SD, %). Bold indicates the best-performing groups that are not significantly different from each other.

Loss	Neuron 2019 Finger	Neuron 2019 Seq.	Neuron 2019 Position	SpanishBCBL Char.	SpanishBCBL Word Top5	SpanishBCBL Word Pos.
$[y]$	60.42 ± 0.57	7.47 ± 4.62	15.79 ± 3.09	49.60 ± 1.07	0.99 ± 0.68	2.03 ± 1.92
$[y + c]$	60.82 ± 0.56	28.50 ± 0.96	16.90 ± 2.31	49.78 ± 1.08	1.38 ± 1.92	1.91 ± 1.91
$[y + e]$	61.52 ± 0.51	7.57 ± 4.19	60.50 ± 0.75	$\mathbf{50.59 \pm 0.93}$	0.93 ± 0.66	33.39 ± 1.36
$[y + c + e]$	$\mathbf{61.91 \pm 0.53}$	28.49 ± 1.93	60.83 ± 0.75	$\mathbf{50.60 \pm 0.80}$	1.38 ± 1.91	33.28 ± 1.44
$[y], [c], [e]$	$\mathbf{62.02 \pm 0.70}$	30.00 ± 1.93	60.95 ± 0.93	$\mathbf{50.44 \pm 0.91}$	1.29 ± 1.96	32.81 ± 1.49

5.4 Held-Out Tests

We next evaluated whether HiLa generalizes across participants and contexts. In a leave-one-participant-out adaptation setting, both alignment-only adaptation, in which the backbone was

frozen and only the participant-specific adapter was trained, and full-backbone fine-tuning improved with increasing numbers of K trials per sequence (Fig. 4a–b). With $K = 5$ trials, performance reached 41.4% accuracy, with consistent performance across held-out participants (Fig. S2). The full hierarchically supervised model consistently matched or exceeded target-only supervision.

Generalization to novel task contexts was then assessed by splitting the data by sequences (Fig. 4c). Adaptation performance reflected the difficulty of the cross-context tasks: unseen order was the hardest condition, whereas unseen timing with the order trained reached higher accuracy. When the sequence was unseen but both order and timing were trained, adaptation achieved higher accuracy. These results are consistent with the hierarchy observed in the neuroscience literature. The hierarchically supervised model consistently outperformed the Target-only model across these cross-context conditions.

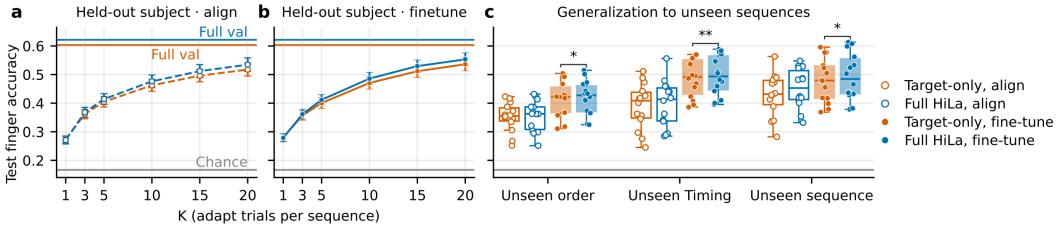

Figure 4: Cross-subject adaptation and sequence generalization of HiLa. **(a–b)** Held-out-subject accuracy with increasing adaptation trials using alignment or fine-tuning. **(c)** Generalization to unseen finger orders, timing patterns, and sequences.

5.5 Branch Splitting Preserves Accuracy While Improving Latent Organization

We next examined how hierarchical supervision organizes the latent representation. Without explicit branch splitting, the latent space showed no clear factor-specific organization: sequence, finger identity, and serial position were substantially intermixed (Fig. 5a). In contrast, branch splitting produced clearly structured and interpretable latent spaces, with each branch exhibiting organization corresponding to its assigned behavioral factor (Fig. 5b), while preserving comparable finger-decoding performance.

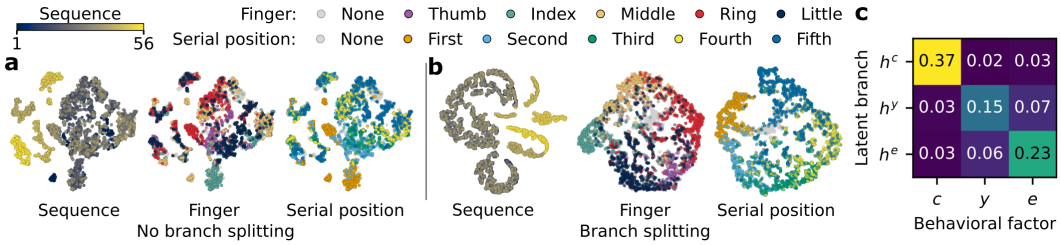

Figure 5: Branch splitting produces factor-specific and more decontextualized representations. **(a–b)** t-SNE projections of the shared latent representation with/without branch splitting, colored by ground truth labels. **(c)** Normalized mutual information matrix.

We then asked whether stronger separation could further improve factorization. Adding global or group-wise decorrelation objectives did not improve decoding, and adversarial objectives produced no consistent gain in the primary finger task (Fig. S1). Their effects were factor-specific: adversarial removal of sequence information from target- or event-related branches improved context decoding in the corresponding comparisons, but adversarial objectives on the target branch did not improve finger decoding. Thus, stronger statistical disentanglement did not translate into better target decoding, suggesting that preserving some shared information across factors may be beneficial rather than enforcing complete independence.

Mean per-dimension mutual-information scores on validation latents showed that, for each branch, the normalized value was largest for the behavioral factor that branch supervises (Fig. 5c). Conditional normalized scores $I(Z; c \mid y)/H(c \mid y)$ and $I(Z; e \mid y)/H(e \mid y)$ were lower when Z was the finger branch alone (0.06 ± 0.01 and 0.09 ± 0.01) than when Z was taken from the one-branch configuration (0.17 ± 0.01 and 0.18 ± 0.01). This reduction indicates that the factor-specific representations retained less information about non-target factors beyond that explained by the target variable, consistent with more decontextualized representations.

5.6 HiLa Learns Biologically Plausible Sequential Patterns

We next examined whether the information exploited by hierarchical supervision was supported by structured, biologically plausible MEG signals (Fig. 6). Task-loss gradient maps showed distinct spatial patterns for finger, sequence, and serial-position supervision (Fig. 6a). Pairwise Pearson and cosine similarities showed a consistent structure: target and event representations were more similar to each other than either was to context, while context was comparatively distinct (Fig. 6b), consistent with the mutual-information analysis.

We further quantified sensor contributions using regional ablations, replacing sensors within predefined regions with Gaussian noise (Fig. 6c,d). Ablations produced larger accuracy decreases for finger and serial-position decoding than for sequence decoding, with generally larger effects in the left hemisphere, consistent with right hand movement. The strongest effects were observed in central, temporal, and frontal regions, while differences between finger and serial-position decoding were most pronounced in central and parietal regions. The prominent contribution of central regions is consistent with the expected involvement of sensorimotor cortex in finger movements. Together, these spatially structured and factor-dependent effects provide converging evidence that HiLa exploits biologically plausible MEG signals rather than nonspecific signal structure.

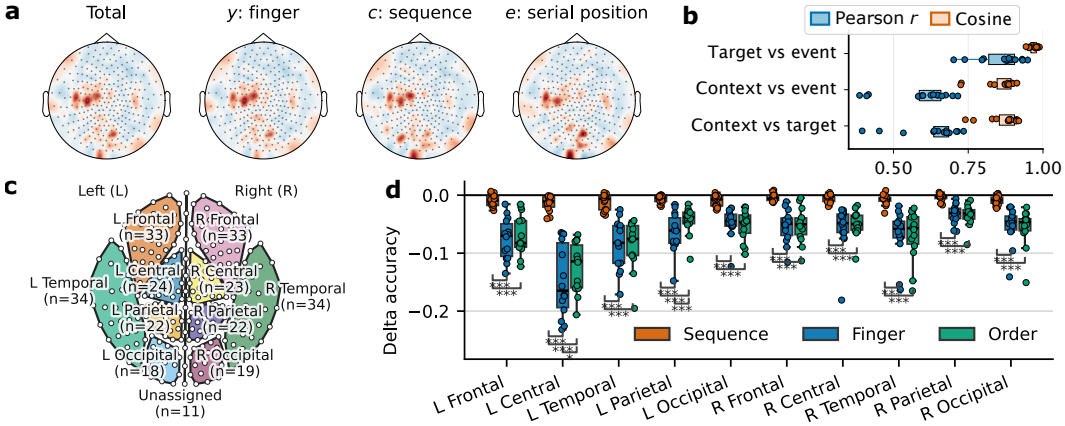

Figure 6: Validation of branch-specific latent structure and sensor-level contributions. **(a)** Sensor-level task-loss gradient maps for a representative participant (repeats averaged). **(b)** Pairwise task-gradient similarity by pearson correlation and cosine similarity. **(c)** Sensor regions used for regional ablation. **(d)** Change in accuracy after replacing each sensor region with Gaussian noise relative to the intact-input baseline; dots denote individual participants.

6 Discussion

HiLa achieves four main results in non-invasive motor decoding. First, supervising a shared encoder with context and event labels improves movement target decoding over target-only training, and the gain holds for held-out participants and held-out contexts. Second, HiLa reaches 62.02%/50.44% accuracy from a single 500 ms MEG window, with no cross window information mixing, sequence prior, or post-hoc correction; its accuracy comes directly from the neural signal. Third, a new participant can be calibrated by training only the participant-specific input stage (210k parameters) while the rest of the model stays frozen, reducing calibration requirements and suggesting a strong encoder backbone (Houlsby et al., 2019). Fourth, branch-specific latents make the representation interpretable at no cost to accuracy: each branch is specialized and decontextualized. Sensor-level analyses show that finger and position decoding rely on largely overlapping left-lateralized sensors, differing mainly in left central and parietal regions.

These properties close a gap in motor decoding. Real motor behavior is sequential: keystroke kinematics depend on upcoming keys (Soechting & Flanders, 1992), and motor areas queue upcoming actions by serial position and separate preparation from execution (Averbeck et al., 2002; Kornysheva et al., 2019; Yokoi & Diedrichsen, 2019). Current non-invasive BCIs decode isolated fingers (Ding et al., 2025) or add cross window context at the output through a language model (Lévy et al., 2025), as do invasive handwriting and speech BCIs (Willett et al., 2023). HiLa focuses on motor signals

only, a required step toward continuous decoding of naturalistic movement (Edelman et al., 2019) that wearable OPM-MEG now makes practical(Boto et al., 2018).

Hierarchical supervision helps by encouraging the encoder to represent structured variation that is relevant to the motor task. Rather than treating context and serial position as nuisance variation, auxiliary targets encourage the shared encoder to capture these factors alongside the movement target (Caruana, 1997; Ruder, 2017). These labels are used only during training, so the model cannot directly use the context label to predict the movement target at test time. We decode individual events from local MEG windows rather than modeling the full motor sequence with a recurrent network (Pandarinath et al., 2018; Sani et al., 2024). This formulation avoids requiring stable long-range sequence modeling, which we found difficult to train on our MEG data.

The two datasets test hierarchical supervision under different conditions. The finger-sequence task (Kornysheva et al., 2019) is highly structured: a small set of memorized five-finger sequences is repeated many times, so each sequence and serial position is well sampled and balanced. The SpanishBCBL typing task (Lévy et al., 2026) is closer to natural behavior but more challenging. Some sentences and words are less frequent, the number of possible keystroke sequences is much larger, and the neural signal reflects linguistic information at multiple levels in addition to the motor command (Lévy et al., 2026; 2025). In this setting, context supervision provides less consistent benefit, whereas serial position supervision remains beneficial. This difference may partly reflect the amount of data available for learning each contextual factor. Recent work on neural decoding has shown that decoding performance continues to improve with increasing amounts of brain recording per participant (Zhang et al., 2026), following approximately log-linear scaling with training data and with stronger gains from additional data within subjects (Banville et al., 2025). Thus, the limited benefit of context supervision in typing may reflect not only the complexity of the contextual labels, but also insufficient repeated observations to learn them reliably.

The improvement from serial-position supervision is consistent with the role of ordinal structure in sequential motor control: motor and premotor areas organize upcoming actions by their position within a sequence (Averbeck et al., 2002; Kornysheva et al., 2019). Serial position provides a discrete, low-cardinality target that is stable across repetitions, unlike continuous elapsed time, which varies with reaction time, execution speed, and motor tempo. HiLa therefore supervises the ordinal structure of the motor program without directly modeling its temporal dynamics.

Supervised structure outperforms enforced invariance. Branch splitting separates the objectives at the latent level: accuracy is unchanged and the latents specialize. We attribute the unchanged accuracy to a trade-off between reduced interference among objectives (Yu et al., 2020; Standley et al., 2020) and reduced capacity per factor. Enforcing full independence is counterproductive: adversarially removing participant information lowered accuracy. This outcome follow from known properties of adversarial invariance, which is unstable to optimize (Ganin et al., 2016) and costs accuracy when the removed variable is correlated with the target (Zhao et al., 2019). Supervised branches capture this dependency instead of fighting it. The same design contributes to three wider problems. It separates mixed non-invasive sources by the behavioral variables they explain, rather than by statistical independence or an ill-posed inverse model (Baillet, 2017). It supplies the supervision that disentanglement requires(Locatello et al., 2019; 2020), using labels that are naturally available in structured tasks, and shows that partial disentanglement is sufficient for accurate decoding. It also turns the behavioral hierarchy into a principled set of auxiliary tasks (Navon et al., 2021).

The main limitations are data scale and the size of the hierarchical gain. The encoder contributes most of the decoding performance, while hierarchical supervision provides a smaller but consistent improvement, showing that exploiting task structure can further improve decoding. The modest gain also suggests that static factorization may be insufficient to capture the full dynamics mixing of sequential motor behavior. Only two datasets were available, and no benchmark currently exists for non-invasive sequential motor decoding, unlike motor imagery (Chevallier et al., 2024). Larger sequential datasets are needed to determine how the benefit of hierarchical supervision scales. A natural next step is dynamical demixing: modeling how context and serial position evolve across successive events using latent dynamical models (Sani et al., 2024; Vahidi et al., 2025), while adapting these approaches to noisy MEG recordings.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript writing and language polishing, code development and debugging, and LaTeX formatting and build issues. In particular, LLMs were used to generate and modify code for data analysis and visualization, and to improve the clarity, grammar, and concision of the manuscript. All AI-assisted code and text were reviewed, tested, and validated by the authors. LLMs were not used to generate or interpret experimental results, make scientific decisions, or determine the conclusions of the work. The authors take full responsibility for the final code, analyses, scientific claims, and manuscript content.

REFERENCES

Hamdi Altaheri, Ghulam Muhammad, and Mansour Alsulaiman. Physics-informed attention temporal convolutional network for eeg-based motor imagery classification. *IEEE transactions on industrial informatics*, 19(2):2249–2258, 2022.

Hamdi Altaheri, Fakhri Karray, and Amir-Hossein Karimi. Temporal convolutional transformer for eeg based motor imagery decoding. *Scientific Reports*, 15, 09 2025. doi: 10.1038/s41598-025-16219-7.

Bruno B. Averbeck, Matthew V. Chafee, David A. Crowe, and Apostolos P. Georgopoulos. Parallel processing of serial movements in prefrontal cortex. *Proceedings of the National Academy of Sciences*, 99(20):13172–13177, Sept 2002. ISSN 1091-6490. doi: 10.1073/pnas.162485599. URL http://dx.doi.org/10.1073/pnas.162485599.

Sylvain Baillet. Magnetoencephalography for brain electrophysiology and imaging. *Nature neuroscience*, 20(3):327–339, 2017.

Hubert Banville, Yohann Benchetrit, Stéphane d'Ascoli, Jérémy Rapin, and Jean-Rémi King. Scaling laws for decoding images from brain activity, 2025. URL https://arxiv.org/abs/2501.15322.

David Bethge, Philipp Hallgarten, Ozan Özdenizci, Ralf Mikut, Albrecht Schmidt, and Tobias Grosse-Puppendahl. Exploiting multiple eeg data domains with adversarial learning. In *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 3154–3158, 2022. doi: 10.1109/EMBC48229.2022.9871743.

Elena Boto, Niall Holmes, James Leggett, Gillian Roberts, Vishal Shah, Sofie S Meyer, Leonardo Duque Muñoz, Karen J Mullinger, Tim M Tierney, Sven Bestmann, et al. Moving magnetoencephalography towards real-world applications with a wearable system. *Nature*, 555 (7698):657–661, 2018.

Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. *Advances in neural information processing systems*, 29, 2016.

Matthew J Brookes, James Leggett, Molly Rea, Ryan M Hill, Niall Holmes, Elena Boto, and Richard Bowtell. Magnetoencephalography with optically pumped magnetometers (opm-meg): the next generation of functional neuroimaging. *Trends in neurosciences*, 45(8):621–634, 2022.

Nicholas S. Card, Maitreyee Wairagkar, Carrina Iacobacci, Xianda Hou, Tyler Singer-Clark, Francis R. Willett, Erin M. Kunz, Chaofei Fan, Maryam Vahdati Nia, Darrel R. Deo, Aparna Srinivasan, Eun Young Choi, Matthew F. Glasser, Leigh R. Hochberg, Jaimie M. Henderson, Kiarash Shahlaie, Sergey D. Stavisky, and David M. Brandman. An accurate and rapidly calibrating speech neuroprosthesis. *New England Journal of Medicine*, 391(7):609–618, Aug 2024. ISSN 1533-4406. doi: 10.1056/nejmoa2314132. URL http://dx.doi.org/10.1056/NEJMoa2314132.

Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. *Advances in neural information processing systems*, 31, 2018.

Sylvain Chevallier, Igor Carrara, Bruno Aristimunha, Pierre Guetschel, Sara Sedlar, Bruna Lopes, Sebastien Velut, Salim Khazem, and Thomas Moreau. The largest eeg-based bci reproducibility study for open science: the moabb benchmark. *arXiv preprint arXiv:2404.15319*, 2024.

Mark M Churchland, John P Cunningham, Matthew T Kaufman, Justin D Foster, Paul Nuyujukian, Stephen I Ryu, and Krishna V Shenoy. Neural population dynamics during reaching. *Nature*, 487 (7405):51–56, 2012.

Michael Cogswell, Faruk Ahmed, Ross B. Girshick, C. Lawrence Zitnick, and Dhruv Batra. Reducing overfitting in deep networks by decorrelating representations. *CoRR*, abs/1511.06068, 2015. URL https://api.semanticscholar.org/CorpusID:2222933.

Yidan Ding, Chalisa Udompanyawit, Yisha Zhang, and Bin He. Eeg-based brain-computer interface enables real-time robotic hand control at individual finger level. *Nature Communications*, 16, 06 2025. doi: 10.1038/s41467-025-61064-x.

Yidan Ding, Joshua Kosnoff, and Bin He. A holistic perspective on noninvasive brain–computer interfaces. *Trends in Neurosciences*, 49(7):566–581, July 2026. ISSN 0166-2236. doi: 10.1016/j.tins.2026.04.009. URL http://dx.doi.org/10.1016/j.tins.2026.04.009.

Laura N. Driscoll, Lea Duncker, and Christopher D. Harvey. Representational drift: Emerging theories for continual learning and experimental future directions. *Current Opinion in Neurobiology*, 76: 102609, 2022. doi: 10.1016/j.conb.2022.102609.

Alexandre Défossez, Charlotte Caucheteux, Jérémy Rapin, Ori Kabeli, and Jean-Rémi King. Decoding speech perception from non-invasive brain recordings. *Nature Machine Intelligence*, 5(10):1097–1107, Oct 2023. ISSN 2522-5839. doi: 10.1038/s42256-023-00714-5. URL http://dx.doi.org/10.1038/s42256-023-00714-5.

B. J. Edelman, J. Meng, D. Suma, C. Zurn, E. Nagarajan, B. S. Baxter, C. C. Cline, and B. He. Noninvasive neuroimaging enhances continuous neural tracking for robotic device control. *Science Robotics*, 4(31), June 2019. ISSN 2470-9476. doi: 10.1126/scirobotics.aaw6844. URL http://dx.doi.org/10.1126/scirobotics.aaw6844.

Bradley J. Edelman, Shuailei Zhang, Gerwin Schalk, Peter Brunner, Gernot Müller-Putz, Cuntai Guan, and Bin He. Non-invasive brain-computer interfaces: State of the art and trends. *IEEE Reviews in Biomedical Engineering*, 18:26–49, 2025. ISSN 1941-1189. doi: 10.1109/rbme.2024.3449790. URL http://dx.doi.org/10.1109/RBME.2024.3449790.

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.

Daniel M Goldenholz, Seppo P Ahlfors, Matti S Hämäläinen, Dahlia Sharon, Mamiko Ishitobi, Lucia M Vaina, and Steven M Stufflebeam. Mapping the signal-to-noise-ratios of cortical sources in magnetoencephalography and electroencephalography. *Human brain mapping*, 30(4):1077–1086, 2009.

Adam Gosztolai, Robert L. Peach, Alexis Arnaudon, Mauricio Barahona, and Pierre Vandergheynst. Marble: interpretable representations of neural population dynamics using geometric deep learning. *Nature Methods*, 22(3):612–620, Feb 2025. ISSN 1548-7105. doi: 10.1038/s41592-024-02582-2. URL http://dx.doi.org/10.1038/s41592-024-02582-2.

Joachim Gross. Magnetoencephalography in cognitive neuroscience: a primer. *Neuron*, 104(2): 189–204, 2019.

Joachim Gross, Sylvain Baillet, Gareth R. Barnes, Richard N. Henson, Arjan Hillebrand, Ole Jensen, Karim Jerbi, Vladimir Litvak, Burkhard Maess, Robert Oostenveld, Lauri Parkkonen, Jason R. Taylor, Virginie van Wassenhove, Michael Wibral, and Jan-Mathijs Schoffelen. Good practice for conducting and reporting meg research. *NeuroImage*, 65:349–363, 2013. ISSN 1053-8119. doi: https://doi.org/10.1016/j.neuroimage.2012.10.001. URL https://www.sciencedirect.com/science/article/pii/S1053811912009895.

Emma L. Hall, Siân E. Robson, Peter G. Morris, and Matthew J. Brookes. The relationship between meg and fmri. *NeuroImage*, 102:80–91, Nov 2014. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2013.11.005. URL http://dx.doi.org/10.1016/j.neuroimage.2013.11.005.

Matti Hämäläinen, Riitta Hari, Risto J Ilmoniemi, Jukka Knuutila, and Olli V Lounasmaa. Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain. *Reviews of modern Physics*, 65(2):413, 1993.

Mo Han, Ozan Ozdenizci, Ye Wang, Toshiaki Koike-Akino, and Deniz Erdogmus. Disentangled adversarial autoencoder for subject-invariant physiological feature extraction. *IEEE Signal Processing Letters*, 27:1565–1569, 2020. ISSN 1558-2361. doi: 10.1109/lsp.2020.3020215. URL http://dx.doi.org/10.1109/LSP.2020.3020215.

Riitta Hari, Lauri Parkkonen, and Cathy Nangini. The brain in time: insights from neuromagnetic recordings. *Annals of the New York Academy of Sciences*, 1191(1):89–109, Mar 2010. ISSN 1749-6632. doi: 10.1111/j.1749-6632.2010.05438.x. URL http://dx.doi.org/10.1111/j.1749-6632.2010.05438.x.

He He and Dongrui Wu. Transfer learning for brain–computer interfaces: A euclidean space data alignment approach. *IEEE Transactions on Biomedical Engineering*, 67(2):399–410, Feb 2020. ISSN 1558-2531. doi: 10.1109/tbme.2019.2913914. URL http://dx.doi.org/10.1109/TBME.2019.2913914.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Sy2fzU9gl.

Jonathan Ho, Nal Kalchbrenner, Dirk Weissenborn, and Tim Salimans. Axial attention in multidimensional transformers, 2020. URL https://openreview.net/forum?id=H1e5GJBtDr.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.

Cole Hurwitz, Akash Srivastava, Kai Xu, Justin Jude, Matthew Perich, Lee Miller, and Matthias Hennig. Targeted neural dynamical modeling. *Advances in Neural Information Processing Systems*, 34:29379–29392, 2021.

Changhong Jing, Yan Liu, Shuqiang Wang, Bruce XB Yu, Gong Chen, Zhejing Hu, Zhi Zhang, and Yanyan Shen. Ptsm: Physiology-aware and task-invariant spatio-temporal modeling for cross-subject eeg decoding. *arXiv preprint arXiv:2508.11357*, 2025.

Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2649–2658. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/kim18b.html.

Dominik Klepl, Min Wu, and Fei He. Graph neural network-based eeg classification: A survey. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 32:493–503, 2024.

Dmitry Kobak, Wieland Brendel, Christos Constantinidis, Claudia E Feierstein, Adam Kepecs, Zachary F Mainen, Xue-Lian Qi, Ranulfo Romo, Naoshige Uchida, and Christian K Machens. Demixed principal component analysis of neural population data. *elife*, 5:e10989, 2016.

Katja Kornysheva and Jörn Diedrichsen. Human premotor areas parse sequences into their spatial and temporal features. *eLife*, 3, Aug 2014. ISSN 2050-084X. doi: 10.7554/elife.03043. URL http://dx.doi.org/10.7554/eLife.03043.

Katja Kornysheva, Daniel Bush, Sofie S Meyer, Anna Sadnicka, Gareth Barnes, and Neil Burgess. Neural competitive queuing of ordinal structure underlies skilled sequential action. *Neuron*, 101 (6):1166–1180, 2019.

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.

Jarod Lévy, Mingfang Zhang, Svetlana Pinet, Jérémy Rapin, Hubert Banville, Stéphane d'Ascoli, and Jean-Rémi King. Brain-to-text decoding: A non-invasive approach via typing. *arXiv preprint arXiv:2502.17480*, 2025.

Jarod Lévy, Mingfang Zhang, Svetlana Pinet, Jérémy Rapin, Hubert Banville, Stéphane d'Ascoli, and Jean-Rémi King. Noninvasive decoding of typed sentences from human brain activity. *Nature Neuroscience*, pp. 1–7, 2026.

Xiaoxiao Li, Yuan Zhou, Nicha Dvornek, Muhan Zhang, Siyuan Gao, Juntang Zhuang, Dustin Scheinost, Lawrence H Staib, Pamela Ventola, and James S Duncan. Braingnn: Interpretable brain graph neural network for fmri analysis. *Medical image analysis*, 74:102233, 2021.

Francesco Locatello, Stefan Bauer, Mario Lučić, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Frederic Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning*, 2019. URL http://proceedings.mlr.press/v97/locatello19a.html. Best Paper Award.

Francesco Locatello, Ben Poole, Gunnar Raetsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6348–6359. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/locatello20a.html.

Nikos K. Logothetis. What we can do and what we cannot do with fmri. *Nature*, 453(7197):869–878, June 2008. ISSN 1476-4687. doi: 10.1038/nature06976. URL http://dx.doi.org/10.1038/nature06976.

Bruna J. Lopes, Gabriel Schwartz, Sylvain Chevallier, Raphael Y. de Camargo, and Bruno Aristimunha. Learning aligned eeg representations with subject-specific encoders. *Scientific Reports*, Sept 2026. ISSN 2045-2322. doi: 10.1038/s41598-026-69220-z. URL http://dx.doi.org/10.1038/s41598-026-69220-z.

Fernando Lopes da Silva. Eeg and meg: Relevance to neuroscience. *Neuron*, 80(5):1112–1128, Dec 2013. ISSN 0896-6273. doi: 10.1016/j.neuron.2013.10.017. URL http://dx.doi.org/10.1016/j.neuron.2013.10.017.

Valerio Mante, David Sussillo, Krishna V Shenoy, and William T Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *nature*, 503(7474):78–84, 2013.

Apolline Mellot, Antoine Collas, Pedro L. C. Rodrigues, Denis Engemann, and Alexandre Gramfort. Harmonizing and aligning m/eeg datasets with covariance-based techniques to enhance predictive regression modeling. *Imaging Neuroscience*, 1, Dec 2023. ISSN 2837-6056. doi: 10.1162/imag_a_00040. URL http://dx.doi.org/10.1162/imag_a_00040.

Sean L. Metzger, Kaylo T. Littlejohn, Alexander B. Silva, David A. Moses, Margaret P. Seaton, Ran Wang, Maximilian E. Dougherty, Jessie R. Liu, Peter Wu, Michael A. Berger, Inga Zhuravleva, Adelyn Tu-Chan, Karunesh Ganguly, Gopala K. Anumanchipalli, and Edward F. Chang. A high-performance neuroprosthesis for speech decoding and avatar control. *Nature*, 620(7976):1037–1046, Aug 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06443-4. URL http://dx.doi.org/10.1038/s41586-023-06443-4.

Aviv Navon, Idan Achituve, Haggai Maron, Gal Chechik, and Ethan Fetaya. Auxiliary learning by implicit differentiation. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=n7wIfYPdVet.

Ozan Özdenizci, Ye Wang, Toshiaki Koike-Akino, and Deniz Erdoğmuş. Learning invariant representations from eeg via adversarial inference. *IEEE access*, 8:27074–27085, 2020.

Chethan Pandarinath, K Cora Ames, Abigail A Russo, Ali Farshchian, Lee E Miller, Eva L Dyer, and Jonathan C Kao. Latent factors and dynamics in motor cortex and their application to brain–machine interfaces. *The Journal of Neuroscience*, 38(44):9390–9401, 2018.

Arthur Pellegrino, Heike Stein, and N. Alex Cayco-Gajic. Dimensionality reduction beyond neural subspaces with slice tensor component analysis. *Nature Neuroscience*, 27(6):1199–1210, jun 2024. ISSN 1546-1726. doi: 10.1038/s41593-024-01626-2. URL https://www.nature.com/articles/s41593-024-01626-2. Publisher: Nature Publishing Group TLDR: It is argued that task-relevant variability in neural data can also cofluctuate over trials or time, defining distinct 'covariability classes' that may co-occur within the same dataset, and developed sliceTCA, a new unsupervised dimensionality reduction method for neural data tensors.

Maria Carla Piastra, Andreas Nüßing, Johannes Vorwerk, Maureen Clerc, Christian Engwer, and Carsten H. Wolters. A comprehensive study on electroencephalography and magnetoencephalography sensitivity to cortical and subcortical sources. *Human Brain Mapping*, 42(4):978–992, Nov 2020. ISSN 1097-0193. doi: 10.1002/hbm.25272. URL http://dx.doi.org/10.1002/hbm.25272.

Herbert Ramoser, Johannes Muller-Gerking, and Gert Pfurtscheller. Optimal spatial filtering of single trial eeg during imagined hand movement. *IEEE transactions on rehabilitation engineering*, 8(4): 441–446, 2000.

Soheil Rayatdoost, Yufeng Yin, David Rudrauf, and Mohammad Soleymani. Subject-invariant eeg representation learning for emotion recognition. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3955–3959. IEEE, 2021.

Sebastian Ruder. An overview of multi-task learning in deep neural networks, 2017. URL https://arxiv.org/abs/1706.05098.

Michael E. Rule, Timothy O'Leary, and Christopher D. Harvey. Causes and consequences of representational drift. *Current Opinion in Neurobiology*, 58:141–147, 2019. doi: 10.1016/j.conb.2019.08.005.

Abigail A Russo, Sean R Bittner, Sean M Perkins, Jeffrey S Seely, Brian M London, Antonio H Lara, Andrew Miri, Najja J Marshall, Adam Kohn, Thomas M Jessell, et al. Motor cortex embeds muscle-like commands in an untangled population response. *Neuron*, 97(4):953–966, 2018.

Victor Sanh, Thomas Wolf, and Sebastian Ruder. A hierarchical multi-task approach for learning embeddings from semantic tasks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6949–6956, July 2019. ISSN 2159-5399. doi: 10.1609/aaai.v33i01.33016949. URL http://dx.doi.org/10.1609/aaai.v33i01.33016949.

Omid G Sani, Hamidreza Abbaspourazad, Yan T Wong, Bijan Pesaran, and Maryam M Shanechi. Modeling behaviorally relevant neural dynamics enabled by preferential subspace identification. *Nature neuroscience*, 24(1):140–149, 2021.

Omid G Sani, Bijan Pesaran, and Maryam M Shanechi. Dissociative and prioritized modeling of behaviorally relevant neural dynamics using recurrent neural networks. *Nature neuroscience*, 27 (10):2033–2045, 2024.

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, Aug 2017. ISSN 1097-0193. doi: 10.1002/hbm.23730. URL http://dx.doi.org/10.1002/hbm.23730.

Steffen Schneider, Jin Hwa Lee, and Mackenzie Weygandt Mathis. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 617(7960):360–368, 2023.

Jan-Mathijs Schoffelen and Joachim Gross. Source connectivity analysis with meg and eeg. *Human Brain Mapping*, 30(6):1857–1865, Feb 2009. ISSN 1097-0193. doi: 10.1002/hbm.20745. URL http://dx.doi.org/10.1002/hbm.20745.

Paul Steven Scotti, Mihir Tripathy, Cesar Torrico, Reese Kneeland, Tong Chen, Ashutosh Narang, Charan Santhirasegaran, Jonathan Xu, Thomas Naselaris, Kenneth A. Norman, and Tanishq Mathew Abraham. Mindeye2: Shared-subject models enable fMRI-to-image with 1 hour of data. In *ICLR 2024 Workshop on Representational Alignment*, 2024. URL https://openreview.net/forum?id=paqqd1O0Dl.

Xinke Shen, Xianggen Liu, Xin Hu, Dan Zhang, and Sen Song. Contrastive learning of subject-invariant eeg representations for cross-subject emotion recognition. *IEEE Transactions on Affective Computing*, 14(3):2496–2511, July 2023. ISSN 2371-9850. doi: 10.1109/taffc.2022.3164516. URL http://dx.doi.org/10.1109/TAFFC.2022.3164516.

Krishna Shenoy and Jonathan Kao. Measurement, manipulation and modeling of brain-wide neural population dynamics. *Nature Communications*, 12, 12 2021. doi: 10.1038/s41467-020-20371-1.

J. F. Soechting and M. Flanders. Organization of sequential typing movements. *Journal of Neurophysiology*, 67(5):1275–1290, May 1992. ISSN 1522-1598. doi: 10.1152/jn.1992.67.5.1275. URL http://dx.doi.org/10.1152/jn.1992.67.5.1275.

Anders Søgaard and Yoav Goldberg. Deep multi-task learning with low level tasks supervised at lower layers. In Katrin Erk and Noah A. Smith (eds.), *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 231–235, Berlin, Germany, aug 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-2038. URL https://aclanthology.org/P16-2038/.

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, 2023. ISSN 1558-0210. doi: 10.1109/tnsre.2022.3230250. URL http://dx.doi.org/10.1109/TNSRE.2022.3230250.

Trevor Standley, Amir Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9120–9132. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/standley20a.html.

Parsa Vahidi, Omid G. Sani, and Maryam M. Shanechi. Modeling and dissociation of intrinsic and input-driven neural population dynamics underlying behavior. *Proceedings of the National Academy of Sciences*, 121(7), Feb 2024. ISSN 1091-6490. doi: 10.1073/pnas.2212887121. URL http://dx.doi.org/10.1073/pnas.2212887121.

Parsa Vahidi, Omid G. Sani, and Maryam Shanechi. BRAID: Input-driven nonlinear dynamical modeling of neural-behavioral data. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=3usdM1AuI3.

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, Krishna V. Shenoy, and Jaimie M. Henderson. A high-performance speech neuroprosthesis. *Nature*, 620 (7976):1031–1036, Aug 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06377-x. URL http://dx.doi.org/10.1038/s41586-023-06377-x.

Alex H. Williams, Tony Hyun Kim, Forea Wang, Saurabh Vyas, Stephen I. Ryu, Krishna V. Shenoy, Mark Schnitzer, Tamara G. Kolda, and Surya Ganguli. Unsupervised discovery of demixed, low-dimensional neural dynamics across multiple timescales through tensor component analysis. *Neuron*, 98(6):1099–1115.e8, June 2018. ISSN 0896-6273. doi: 10.1016/j.neuron.2018.05.015. URL http://dx.doi.org/10.1016/j.neuron.2018.05.015.

Atsushi Yokoi and Jörn Diedrichsen. Neural organization of hierarchical motor sequence representations in the human neocortex. *Neuron*, 103(6):1178–1190.e7, Sept 2019. ISSN 0896-6273. doi: 10.1016/j.neuron.2019.06.017. URL http://dx.doi.org/10.1016/j.neuron.2019.06.017.

Byron M. Yu, John P. Cunningham, Gopal Santhanam, Stephen I. Ryu, Krishna V. Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *Journal of Neurophysiology*, 102(1):614–635, July 2009. ISSN 1522-1598. doi: 10.1152/jn.90941.2008. URL http://dx.doi.org/10.1152/jn.90941.2008.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33: 5824–5836, 2020.

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, May 2018. ISSN 1558-2531. doi: 10.1109/tbme.2017.2742541. URL http://dx.doi.org/10.1109/TBME.2017.2742541.

Ce Zhang, Young-Keun Kim, and Azim Eskandarian. Eeg-inception: an accurate and robust end-to-end neural network for eeg-based motor imagery classification. *Journal of Neural Engineering*, 18 (4):046014, 2021.

Mingfang Zhang, Jarod Lévy, Cedric Rommel, Jérémy Rapin, Corentin Bel, Julie Bonnaire, Daniel Nieto, Pierre Bourdillon, Svetlana Pinet, Stéphane d'Ascoli, et al. Accurate decoding of natural sentences from non-invasive brain recordings. *arXiv preprint arXiv:2608.18114*, 2026.

Shu Zhang, Ran Xu, Caiming Xiong, and Chetan Ramaiah. Use all the labels: A hierarchical multi-label contrastive learning framework. In *CVPR*, 2022.

Yu Zhang and Qiang Yang. A survey on multi-task learning, 2021. URL `https://arxiv.org/abs/1707.08114`.

Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant representations for domain adaptation. In *International conference on machine learning*, pp. 7523–7532. PMLR, 2019.

A Additional Results

A.1 Adversarial and Decorrelation Objectives Reveal Asymmetric Dependencies

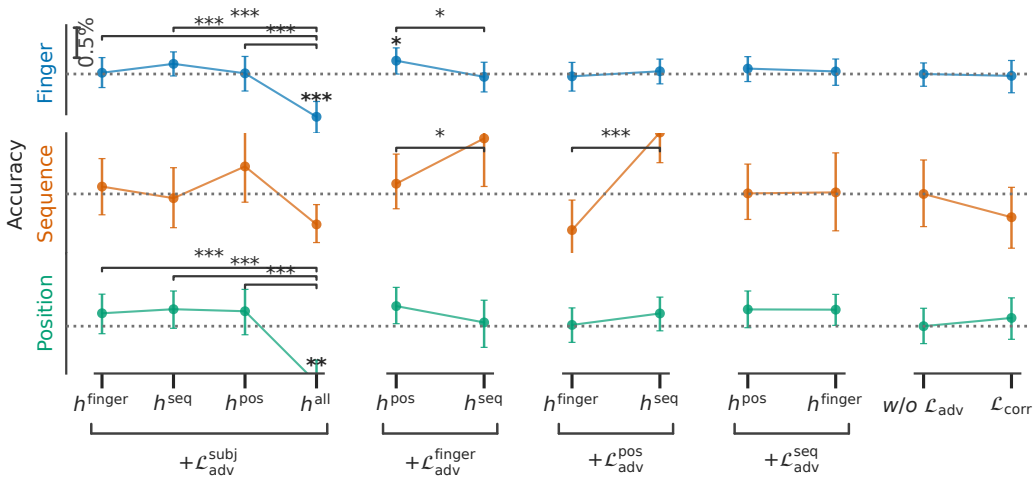

Figure S1: Effects of adversarial and decorrelation objectives on decoding accuracy of Neuron2019. Each group compares the baseline model with an additional adversarial objective applied to the indicated latent representation. The rows show decoding accuracy for finger identity, sequence identity, and serial position, respectively. In the first group, adversarial supervision is applied to the full representation to suppress participant identity, with models trained using different combinations of behavioral supervision. The subsequent groups apply adversarial objectives to suppress finger, serial-position, or sequence information from the corresponding latent branches. The final two conditions apply decorrelation objectives to the latent representation, using within-representation decorrelation ($\mathcal{L}$corr) and cross-branch decorrelation ($\mathcal{L}$cross). Points show mean accuracy across participants and error bars indicate standard deviation. The dotted horizontal line denotes the corresponding baseline performance. Asterisks above individual points indicate comparisons with the baseline, while bracketed asterisks indicate pairwise comparisons within each adversarial group. * $p < .05$, ** $p < .01$, *** $p < .001$, FDR-BH.

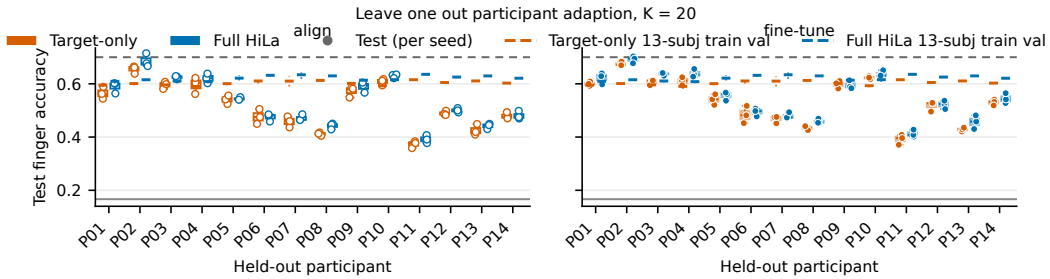

Figure S2: Leave-one-participant-out adaptation using $K = 20$ trials per sequence for adaptation (Neuron2019). Each panel shows results across held-out participants for the two adaptation strategies, alignment (left) and fine-tuning (right). Dashed horizontal lines indicate the corresponding 13-participant training baseline on the validation set, while dots indicate accuracy obtained after adaptation for each fold and seed. Orange and blue denote the Target-only and Full HiLa models, respectively.

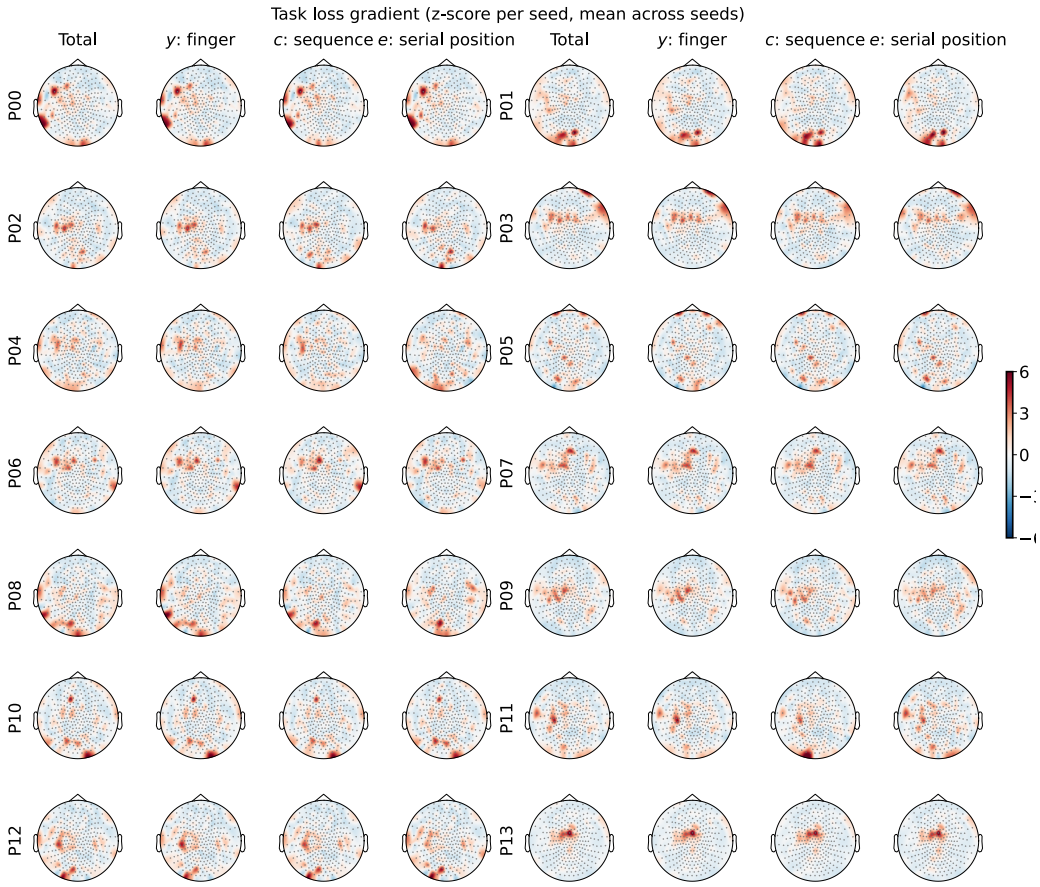

Figure S3: Sensor-level task-loss gradient maps for all participants. Rows correspond to participants, and columns show gradients for the total, finger, sequence, and position losses. Gradients were computed over validation windows, rectified, z-scored, and averaged across seeds. Maps provide measurement-level diagnostics of task-relevant sensor contributions and should not be interpreted as cortical source localization.

Regional sensor-ablation statistics (Neuron2019). We quantified regional sensor contributions by replacing sensors within each predefined region with Gaussian noise and measuring the change in decoding accuracy relative to the intact-input baseline. We then compared ablation-induced drops across heads within each region using paired Wilcoxon signed-rank tests with Holm correction across the three pairwise comparisons per region (Table 4). We additionally ranked regions by their mean accuracy drop for each decoding head (Table 3).

Table 3: Regional sensor-ablation impact for each decoding head. Regions are ranked by mean accuracy drop after replacing sensors in that region with Gaussian noise. More negative values indicate stronger impairment relative to the intact-input baseline. Values are reported in percentage.

Rank	Sequence region	Δ Acc.	Finger region	Δ Acc.	Position region	Δ Acc.
1	Left central	-1.40	Left central	-14.81	Left central	-12.54
2	Left temporal	-1.18	Left temporal	-8.58	Left temporal	-8.49
3	Left occipital	-0.98	Left frontal	-7.57	Left frontal	-6.98
4	Left frontal	-0.92	Right temporal	-6.80	Right temporal	-6.88
5	Right occipital	-0.89	Left parietal	-6.43	Right occipital	-5.59
6	Right temporal	-0.81	Right central	-5.69	Left occipital	-5.26
7	Left parietal	-0.67	Right frontal	-5.37	Right frontal	-5.19
8	Right central	-0.46	Right occipital	-5.28	Right central	-4.91
9	Right frontal	-0.42	Left occipital	-4.77	Left parietal	-4.33
10	Right parietal	-0.41	Right parietal	-3.24	Right parietal	-3.50

Table 4: Within-region pairwise comparisons of sensor-ablation drops across decoding heads. Values are FDR-BH corrected p-values from paired Wilcoxon signed-rank tests (three contrasts per region). $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.

Region	Seq–Fin	Seq–Ord	Fin–Ord	Region	Seq–Fin	Seq–Ord	Fin–Ord
Left central	0.00018^{***}	0.00018^{***}	0.0166^{*}	Right central	0.00018^{***}	0.00018^{***}	0.217
Left frontal	0.00018^{***}	0.00018^{***}	0.153	Right frontal	0.00018^{***}	0.00018^{***}	0.761
Left occipital	0.00018^{***}	0.00018^{***}	0.463	Right occipital	0.00018^{***}	0.00018^{***}	0.104
Left parietal	0.00018^{***}	0.00018^{***}	0.00037^{***}	Right parietal	0.00018^{***}	0.00018^{***}	0.426
Left temporal	0.00018^{***}	0.00018^{***}	0.626	Right temporal	0.00018^{***}	0.00018^{***}	0.715

Table 5: Validation accuracy by finger identity and serial position on the Neuron 2019 dataset. Values are mean ± SD across 5 fold x 4 seeds runs.

Finger identity		Serial position	
Class	Accuracy (%)	Class	Accuracy (%)
Non-press	53.34 ± 2.93	Non-press	52.66 ± 2.73
Thumb	62.78 ± 3.46	First	84.82 ± 2.18
Index	68.21 ± 3.01	Second	61.65 ± 2.88
Middle	68.39 ± 3.91	Third	51.50 ± 3.70
Ring	60.56 ± 4.01	Fourth	51.17 ± 3.92
Little	58.85 ± 2.85	Fifth	64.35 ± 5.17

Table 6: Test-set accuracy by key identity and serial position on the Spanish BCBL dataset. Values are mean ± SD across 20 independent train/validation/test splits. n denotes the mean number of test samples per class across splits.

Key identity			Serial position		
Key	n	Accuracy (%)	Position	n	Accuracy (%)
Space	2572	73.84 ± 2.55	0	3076	71.88 ± 3.21
a	2446	57.82 ± 3.08	1	3010	57.13 ± 2.86
b	201	11.50 ± 4.56	2	2399	37.98 ± 3.75
c	1001	50.49 ± 4.97	3	1778	14.16 ± 5.08
d	547	19.35 ± 4.30	4	1664	4.67 ± 3.13
e	1829	56.19 ± 3.44	5	1487	10.77 ± 4.07
f	126	2.10 ± 2.75	6	1171	13.99 ± 4.30
g	232	10.48 ± 4.64	7	811	0.43 ± 0.65
h	23	0.00 ± 0.00	8	590	0.76 ± 0.97
i	1237	44.67 ± 4.61	9	372	0.49 ± 0.91
j	54	0.00 ± 0.00	10	223	0.24 ± 0.78
k	15	1.94 ± 4.47	11	116	0.00 ± 0.00
l	1640	62.72 ± 4.01	12	65	0.00 ± 0.00
m	382	19.97 ± 5.91	13	13	0.00 ± 0.00
n	1058	48.60 ± 3.08	14	4	0.00 ± 0.00
o	1460	45.40 ± 3.81	15	2	0.00 ± 0.00
p	400	26.99 ± 4.25	16–39	1	0.00 ± 0.00
q	40	7.87 ± 6.72			
r	1125	39.29 ± 5.03			
s	1763	52.34 ± 4.04			
t	590	53.43 ± 4.47			
u	372	37.40 ± 6.89			
v	162	6.34 ± 3.68			
w	9	21.44 ± 20.42			
x	47	0.74 ± 1.84			
y	20	0.12 ± 0.56			
z	20	0.00 ± 0.00			

Table 7: Participant-level validation accuracy for sequence identity, finger identity, and serial position on the Neuron 2019 dataset. Values are mean ± SD across evaluation runs; n denotes the mean number of validation windows per participant.

	Sequence identity		Finger identity		Serial position	
Participant	n	Accuracy (%)	n	Accuracy (%)	n	Accuracy (%)
p01	1464	33.30 ± 3.73	1464	73.05 ± 1.96	1464	71.25 ± 2.65
p02	727	38.07 ± 6.77	727	70.32 ± 3.34	727	63.27 ± 2.70
p03	1460	32.49 ± 3.27	1460	71.10 ± 1.77	1460	70.35 ± 1.82
p04	1152	30.09 ± 3.27	1152	72.58 ± 1.63	1152	66.35 ± 2.12
p05	1238	32.54 ± 4.11	1238	63.11 ± 2.34	1238	63.54 ± 3.05
p06	1419	24.86 ± 1.91	1419	56.36 ± 2.42	1419	56.63 ± 2.76
p07	1407	31.58 ± 3.95	1407	52.63 ± 1.80	1407	60.11 ± 2.59
p08	1210	29.38 ± 2.59	1210	54.45 ± 2.66	1210	52.52 ± 2.56
p09	1448	31.84 ± 3.29	1448	68.19 ± 2.31	1448	67.92 ± 1.44
p10	1434	31.48 ± 3.88	1434	72.32 ± 1.85	1434	69.69 ± 2.97
p11	1320	26.69 ± 2.54	1320	46.15 ± 2.32	1320	47.46 ± 3.21
p12	1364	25.39 ± 2.76	1364	57.79 ± 2.95	1364	56.33 ± 3.37
p13	885	30.91 ± 1.41	885	48.32 ± 3.16	885	44.49 ± 3.01
p14	1441	25.29 ± 2.17	1441	59.69 ± 3.39	1441	56.55 ± 3.15

Table 8: Participant-level test accuracy on the Spanish BCBL dataset. Values are mean ± SD across 20 independent train/validation/test splits. n denotes the mean number of test windows per participant.

Participant	n	Word top-5 (%)	Finger/key (%)	Serial position (%)
p01	1,760	1.45 ± 2.10	45.53 ± 1.90	32.43 ± 1.73
p02	987	1.22 ± 1.79	50.11 ± 2.03	30.24 ± 2.01
p03	492	1.25 ± 2.04	48.07 ± 2.31	30.96 ± 2.50
p04	1,012	1.29 ± 1.95	62.48 ± 1.60	38.85 ± 2.50
p05	499	1.33 ± 2.02	63.11 ± 2.08	36.26 ± 2.60
p06	995	1.27 ± 1.94	54.71 ± 2.06	28.34 ± 1.49
p07	1,000	1.32 ± 1.99	51.44 ± 2.59	32.14 ± 1.99
p08	996	1.32 ± 1.99	43.98 ± 1.62	30.60 ± 1.83
p09	788	1.25 ± 1.96	41.49 ± 2.32	29.36 ± 1.65
p10	1,011	1.35 ± 2.04	56.18 ± 2.16	34.82 ± 2.58
p11	981	1.36 ± 2.09	48.50 ± 1.97	33.44 ± 2.21
p12	1,023	1.29 ± 1.94	52.68 ± 1.38	37.11 ± 1.83
p13	701	1.28 ± 2.07	41.57 ± 2.21	27.43 ± 2.22
p14	1,758	1.24 ± 1.95	51.59 ± 1.52	31.23 ± 1.65
p15	1,992	1.23 ± 1.89	41.09 ± 1.46	31.92 ± 1.87
p16	812	1.31 ± 1.93	63.48 ± 1.94	36.90 ± 2.40
p17	1,021	1.25 ± 1.88	55.87 ± 1.60	38.51 ± 2.30
p18	797	1.35 ± 1.96	49.17 ± 1.94	29.48 ± 1.89
p19	731	1.16 ± 1.87	52.36 ± 1.54	34.37 ± 1.50

B Additional Experimental Details

B.1 Data Details

B.1.1 Neuron2019

This study used published MEG data from Kornysheva et al. (2019).

Memory-guided finger sequence task. Participants performed memorized five-finger tapping sequences from a visual cue. Each trial began with an abstract cue indicating which sequence to produce, followed by a variable delay and a Go cue. After the Go cue, participants executed the sequence from memory, without further visual guidance. All sequences contained one press of each finger, but differed in the position of finger presses and in the timing between presses, forming a 2×2 design over finger position and temporal interval position. Each participant performed four learned sequences, with 60 repetitions per sequence.

In Neuron2019, each participant learned four randomly generated sequences arranged in a 2×2 factorial design. Because exact sequence configurations were not repeated across participants, sequence identity is inherently coupled with participant identity. We therefore treat sequence identity as a task-context variable rather than a participant-independent factor, and do not interpret the context branch as participant-invariant. Hierarchical supervision is instead intended to encourage the shared representation to capture task-relevant contextual structure alongside movement identity and serial position. The shared factorial structure of finger order and timing nevertheless provides common contextual factors across participants. Thus, our results show that contextual and event-level supervision improves movement decoding, rather than that the learned context representation is independent of participant identity.

Data and window statistics. The analyses included 14 participants with valid behavioral and MEG recordings. Only correctly performed trials were retained. MEG signals were downsampled from 1000 Hz to 100 Hz, and fixed-length sliding windows were extracted from the Go cue to 6 s after Go, covering the full sequence-production period. Windows were extracted with a stride of 40 ms, corresponding to 4 samples at 100 Hz. Finger labels were assigned using the press-window definition described in the main Methods. During training, windows were class-balanced by sampling an equal number of windows from each finger class. Summary trial counts by sequence and window counts by finger are reported in Table 9.

Table 9: Participant-level trial counts by sequence and window counts by finger (14 participants; finger labels require press overlap ≥ 0.8, otherwise F0).

	Trials					Windows						
Participant	Total	Seq1	Seq2	Seq3	Seq4	Total	F0	F1	F2	F3	F4	F5
p01	239	59	60	60	60	32900	26630	1256	1253	1252	1256	1253
p02	118	15	25	47	31	16228	13136	618	612	622	612	628
p03	239	60	59	60	60	32901	26685	1244	1208	1257	1251	1256
p04	204	44	48	57	55	28052	23234	1072	845	895	933	1073
p05	208	53	47	55	53	28569	23314	1069	998	1048	1062	1078
p06	236	59	59	59	59	32472	26410	1242	1082	1250	1240	1248
p07	229	59	57	55	58	31518	25521	1189	1200	1202	1196	1210
p08	206	48	53	54	51	28555	23463	1025	1002	1054	973	1038
p09	236	59	59	59	59	32504	26317	1233	1242	1228	1240	1244
p10	235	59	56	60	60	32354	26225	1225	1214	1230	1227	1233
p11	225	50	57	60	58	30944	25361	1159	1170	1003	1164	1087
p12	222	57	52	57	56	30540	24724	1162	1165	1154	1174	1161
p13	144	31	45	40	28	19789	16030	738	752	750	762	757
p14	234	57	60	58	59	32225	26089	1218	1231	1219	1235	1233
Total	2975	710	737	781	747	409551	333139	15450	14974	15164	15325	15499

B.1.2 SPANISH BCBL TYPING (BRAIN2QWERTY)

This study used published MEG data from the DECOMEG Brain2Qwerty release (Lévy et al., 2025)(`https://github.com/facebookresearch/brain2qwerty/tree/main/brain2qwerty_v1`).

Memorized sentence typing task. Participants read Spanish sentences presented word-by-word (rapid serial visual presentation), then typed each sentence from memory on a QWERTY keyboard after a go cue, without on-screen feedback during typing. Each MEG visit included many unique declarative sentences (5–8 words). We analyzed the public MEG subset after Brain2Qwerty V1 participant merging (19 unique people; same as the published data pipeline).

Data and window statistics. We also followed the published preprocessing and splitting script. MEG was band-pass filtered (0.1–20 Hz) and downsampled to 50 Hz. Fixed-length windows were extracted from -200 ms to $+300$ ms around each key-down (25 samples), with pre-key baseline subtraction and per-window robust scaling (clamped to ± 5). Each window was labeled with the pressed key (27 classes: `a–z` and space). One *trial* is one sentence production instance (unique `trial_id` within a session slot). Table 10 reports per-participant trial counts by MEG session and window counts by session and key type (space vs. letter). The full cache contains **190,480** keystroke windows from **5,227** sentence trials across **306** MEG channels.

Table 10: Participant-level sentence-trial counts by MEG session and keystroke-window counts (19 merged participants; 27 QWERTY keys including space).

	Trials (sentences)			Windows (keystrokes)				
Participant	Total	Sess1	Sess2	Total	Sess1	Sess2	Space	Letters
p01	502	241	261	17292	7526	9766	2321	14971
p02	262	130	132	9895	4893	5002	1301	8594
p03	196	130	66	6852	4721	2131	896	5956
p04	461	197	264	17205	7239	9966	2541	14664
p05	583	322	261	19636	9737	9899	2571	17065
p06	198	132	66	7967	5003	2964	1070	6897
p07	264	132	132	10099	5092	5007	1381	8718
p08	198	132	66	7897	4985	2912	1045	6852
p09	198	132	66	7189	5047	2142	924	6265
p10	260	128	132	9713	4716	4997	1273	8440
p11	132	132	0	4698	4698	0	623	4075
p12	264	132	132	10031	5021	5010	1310	8721
p13	132	132	0	4933	4933	0	651	4282
p14	264	132	132	9837	5033	4804	1311	8526
p15	261	132	129	9867	5003	4864	1291	8576
p16	260	128	132	7636	2899	4737	1048	6588
p17	264	132	132	9924	4929	4995	1305	8619
p18	264	132	132	9729	4831	4898	1291	8438
p19	264	132	132	10080	5023	5057	1316	8764
Total	5227	2860	2367	190480	101329	89151	25469	165011

Table 11: Summary of sentence-type occurrences across the cohort (223 unique sentence types in cache).

Statistic (trials per sentence type, cohort-wide)	Value
Unique sentence types	223
Total sentence trials	5227
Min / median / max trials per type	1 / 38 / 41
Mean trials per type	23.44
Min / median / max participants per type	1 / 19 / 19

Table 12: Keystroke-window counts by key class (27 QWERTY labels; cohort total).

Key	Windows	%	Key	Windows	%
(space)	25469	13.4	n	10820	5.7
a	22410	11.8	o	14975	7.9
b	1910	1.0	p	4243	2.2
c	9292	4.9	q	306	0.2
d	5877	3.1	r	10371	5.4
e	19661	10.3	s	17056	9.0
f	1369	0.7	t	6211	3.3
g	2266	1.2	u	3336	1.8
h	309	0.2	v	1732	0.9
i	11646	6.1	w	93	0.0
j	563	0.3	x	474	0.2
k	142	0.1	y	205	0.1
l	15706	8.2	z	155	0.1
m	3883	2.0			

B.2 Benchmark

EEGNet, ATCNet, and TCFormer use the publicly available PyTorch implementations released with TCFormer (`https://github.com/Altaheri/TCFormer`), with only the input dimensions, number of output classes, and dataset-specific hyperparameters adapted to each dataset. The linear baseline applies temporal mean pooling followed by a linear classification layer. The Brain2Qwerty-style baseline consists of six dilated one-dimensional convolutional blocks with learnable subject-specific sensor scaling. Whereas the original Brain2Qwerty model feeds these representations into a sentence-level Transformer decoder, we connect the convolutional encoder directly to a key/finger classification head (`https://github.com/facebookresearch/brain2qwerty`). All baselines predict a single finger or key class from each MEG window using cross-entropy loss and the Adam optimizer. Model checkpoints are selected based on validation accuracy within the training data, and held-out test data are used only for final evaluation.

Hyperparameter optimization. For the Linear and Brain2Qwerty baselines, we used the published default hyperparameters. For the remaining benchmark models, we performed hyperparameter optimization using the Tree-structured Parzen Estimator (TPE) implemented in Optuna. The search was conducted iteratively, expanding or refining the search range when the best configuration reached a search boundary. We continued the search until validation performance showed no improvement for more than 30 trials. For the Spanish BCBL dataset, each Optuna trial was trained for 10 epochs with early stopping, using validation key-classification accuracy as the optimization objective. After hyperparameter optimization, the selected configuration for each architecture was fixed before the final benchmark evaluation.

Evaluation. After fixing the hyperparameters, all benchmark models were evaluated using the same evaluation protocols as HiLa. For the Neuron 2019 dataset, models were evaluated using grouped five-fold cross-validation with four training seeds. For the Spanish BCBL dataset, models were evaluated using 20 independent train/validation/test splits, with 80%, 10%, and 10% of the data assigned to the respective splits. Final Spanish BCBL performance was measured on the held-out test set using the checkpoint with the highest validation accuracy.

B.3 Architecture Ablations

We ran shape-preserving architecture ablations on the HiLa encoder backbone. Each setting keeps the same grouped 5-fold splits, train seeds, optimizer, supervised losses and task linear heads as the baseline; only the flagged module is altered. When a module is removed, it is replaced by the minimal operation that preserves downstream tensor shapes. The ablation procedure for each module is specified in Table 13)

Table 13: Shape-preserving architecture ablations of the HiLa (full) baseline.

Ablation	Description
Baseline	All architectural components are enabled.
− Part. adapter	Removes the per-participant adapter, passing the aligned MEG directly to the shared encoder without subject-specific channel mapping.
− Ch. mixer	Replaces the learned `LinearChannelMixer` with fixed group averaging. Channels are divided into contiguous groups and averaged at each time point, with no additional learnable parameters.
Single-scale conv	Replaces `MultibandTemporalConv` with a single depthwise temporal convolution ($k = 3$), followed by BatchNorm and ELU. The resulting representation is replicated across bands.
− Branch TCN	Removes the per-branch TCN encoder. Each branch is instead temporally mean-pooled and mapped directly to its task-specific dimension using a single linear layer.
− Multi-scale conv	Removes the temporal convolution entirely. The pooled representation is expanded to the required channel dimension and replicated across bands, with no learned convolutional processing.
− Axial attn	Removes axial transformer. The embedding and token-pooling MLPs are retained, but no attention-based feature mixing is performed.
− Pointwise conv	Removes the post-pooling 1×1 convolution, BatchNorm, and SiLU activation. The flattened band–latent–channel representation is passed through unchanged.
− KL	Set β to 0.

B.4 Decorrelation and Adversarial Regularization

Adversarial and Independence Regularization We test two regularization strategies to encourage separation between branch latents. First, covariance-based decorrelation penalties are applied either within the pooled latent representation or between branch groups, penalizing cross-covariances among the sequence, position, and finger latent blocks (Bousmalis et al., 2016; Hurwitz et al., 2021). Second, adversarial gradient-reversal losses (Ganin et al., 2016) are attached to branch representations to suppress non-target information. For example, sequence latents can be adversarially trained against finger, position, and participant labels, while finger latents can be trained against sequence, position, and participant labels. These objectives test which factors can be separated from each branch and which remain coupled in the learned representation.

To probe whether branch latents capture distinct sources of task information, we optionally add decorrelation and adversarial regularizers to the supervised objective. All experiments in this section use the same three-branch, three-task backbone as the main model. The supervised objective is fixed across settings, with loss weights

$$w_{\text{finger}} = 3, \qquad w_{\text{seq}} = 1, \qquad w_{\text{ord}} = 1.5,$$

Thus, the baseline is the same supervised model without additional decorrelation or adversarial regularization. The experimental settings differ only in the added regularization term, as listed in Table 16.

Latent decorrelation. We consider two covariance-based penalties on time-pooled latent representations. The first penalizes correlations within the pooled full latent vector z, encouraging different latent dimensions to carry less redundant information:

$$\mathcal{L}_{\text{corr}} = \sum_{i \neq j} \text{Cov}(z)_{ij}^2.$$

The second penalizes cross-covariance between branch-specific latent blocks,

$$\mathcal{L}_{\text{gcc}} = \|\text{Cov}(z^{\text{seq}}, z^{\text{ord}})\|_F^2 + \|\text{Cov}(z^{\text{seq}}, z^{\text{finger}})\|_F^2 + \|\text{Cov}(z^{\text{ord}}, z^{\text{finger}})\|_F^2.$$

This group cross-covariance penalty directly discourages shared variation between the sequence, position, and finger latent subspaces.

Participant-adversarial regularization. We test whether participant-specific information can be removed from the learned representation. For global participant-adversarial training, a linear participant classifier is attached to the concatenated latent representation through a gradient reversal layer (GRL) (Ganin et al., 2016). The GRL is the identity in the forward pass but reverses gradients during backpropagation,

$$\mathrm{GRL}_\lambda(u) = u, \qquad \frac{\partial \mathrm{GRL}_\lambda(u)}{\partial u} = -\lambda I,$$

with $\lambda = 1$ in our experiments. This encourages the encoder to reduce participant-predictive information while the participant classifier is trained to predict the participant label.

Branch-adversarial regularization. To test directional dependencies between task factors, we attach adversarial nuisance heads to individual branch condition vectors. For a branch representation $u_k \in \{c^{\mathrm{seq}}, c^{\mathrm{ord}}, c^{\mathrm{finger}}\}$, a nuisance classifier q_k predicts a non-target factor t_k through the GRL:

$$\mathcal{L}_{\mathrm{adv},k} = \ell_k(q_k(\mathrm{GRL}(u_k)), t_k)\,.$$

Categorical nuisances use multiclass cross-entropy, and press-overlap nuisances use binary cross-entropy with logits. In branch-adversarial experiments, we suppress sequence, position, finger, or participant information from selected non-corresponding branches. For example, $\mathrm{finger} \leftarrow \mathrm{position}$ denotes adversarially removing position information from the finger branch.

Total objective. The full objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{sup}} + w_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}} + w_{\mathrm{indep}}\mathcal{L}_{\mathrm{indep}} + \sum_k w_{\mathrm{adv},k}\mathcal{L}_{\mathrm{adv},k},$$

where $\mathcal{L}_{\mathrm{indep}}$ is either $\mathcal{L}_{\mathrm{corr}}$ or $\mathcal{L}_{\mathrm{gcc}}$. Setting a weight to zero disables the corresponding regularizer.

B.5 Held-out cross-participant and context tests

Held-out evaluation tests generalization to a participant excluded from pretraining and subsequently adapted in a second stage. We compare a target-only one-branch model with the full hierarchical supervision and branch splitting model.

Leave-one-participant-out pretraining For each held-out participant s and seed, Phase 1 trains on the other 13 participants, with an 80/20 trial-level train/validation split balanced by participant and memorized sequence. We ran with seeds 0–2, creating 14 x 3 = 42 individual held-out tests. The held-out participant contributes no data to pretraining. Phase 2 initializes the held-out participant's alignment map that approximate identity transformation. We evaluate two adaptation modes: **align-only** (participant adapter only), and **finetune** (adapter and encoder). For each local sequence of participant s, $K \in \{1, 3, 5, 10, 15, 20\}$ trials are used for adaptation, with all remaining trials used for evaluation. Train and evaluation windows are finger-balanced.

Context generalization protocol. To test generalization across memorized sequences, the four local sequences are divided into disjoint adaptation and test sets. The sequences have a factorial design, where for each participant, sequence 0,2/1,3 share the same finger press order, and sequence 0,1/2,3 share the same timing profile. We evaluate unseen order (train 0,2 test 1,3, then train 1,3 test 0, 2), unseen timing (train 0,1 vs. 2,3 and train 2,3 test 0, 1), and unseen sequence (train 0,1,2 vs. 3 and rotate). Within training sequences, 20% of trials are reserved for validation; test sequences are used only for evaluation. Forward and inverse splits are averaged for the order- and timing-focused contrasts. Primary metrics are test finger, sequence, and order accuracy.

B.6 Latent-space visualization

For qualitative visualization of branch latent spaces, we extracted validation-set representations c_i^{seq}, c_i^{finger}, and c_i^{ord} from trained models using all validation sliding windows. A fixed subsample was shared across conditions, retaining up to five trials per participant–sequence pair. Within each trial, windows were subsampled to balance finger presses and non-press windows..

We applied t-SNE separately to each branch using scikit-learn ($n_{\mathrm{components}} = 2$, PCA initialization, `learning_rate=auto`, and the training seed as `random_state`). Perplexity was set

to $\min(30, \max(5, \lfloor (n-1)/3 \rfloor))$. Sequence embeddings used a sequential colormap by global sequence identity, while finger and position embeddings used fixed categorical colors matching the manuscript schematic.

B.7 Mutual information.

We quantified the behavioral information encoded by each branch of HiLa using Shannon mutual information (MI) on held-out validation windows. This complements linear probing by measuring both linear and nonlinear statistical dependence without training additional probe networks.

For each MEG window, the model produces three branch representations: Z_{C} (sequence context), Z_{Y} (finger), and Z_{E} (serial order). Each window is associated with sequence identity c, finger identity y, and press order e. We estimate $I(Z_i; Y_j)$ for all branch–factor pairs $i, j \in \{\mathrm{C}, \mathrm{Y}, \mathrm{E}\}$.

We used the all-14 cohort and the same five-fold splits as the main experiments. MI between continuous multivariate latents and discrete factors was estimated with scikit-learn's `mutual_info_classif` ($k = 3$). The estimator was applied independently to each latent dimension, and the resulting MI values were averaged across dimensions and converted from nats to bits. Results are reported as mean $\pm$ SD across four seeds and five folds. For visualization, MI was normalized by the empirical entropy of the corresponding factor, $I(Z;Y)/H(Y)$.

To assess information about sequence and order beyond finger identity, we estimated $I(Z; c \mid y)$ and $I(Z; e \mid y)$ using a stratified estimator. MI was estimated separately within each finger class and averaged with weights proportional to the number of windows in each class. We evaluated both the concatenation of all three branches and the finger branch alone.

B.8 Task-Loss Gradient Saliency

To characterize the sensor-level input sensitivity of the trained model, we computed gradients of each task loss with respect to the MEG input. Analyses used the all-14 cohort and the five-fold cross-validation splits from the main experiments. For each seed and fold, we loaded the best validation checkpoint, set the model to evaluation mode, and computed gradients on the corresponding validation windows.

For an input window $x \in \mathbb{R}^{T \times C}$, where T denotes the number of time points and C the number of MEG channels, we computed gradients of three task-specific cross-entropy losses: sequence identity (L_{seq}), finger identity (L_{fing}), and serial position (L_{ord}). We additionally computed gradients of the total training objective (L_{tot}). Each loss was backpropagated independently from the same forward pass.

For subject s, the gradient-based sensitivity of channel c for loss L_b was defined as the mean absolute gradient over validation windows and time points:

$$g_{s,c}^{(b)} = \frac{1}{|\mathcal{W}_s|} \sum_{w \in \mathcal{W}_s} \frac{1}{T} \sum_{t=1}^{T} \left| \frac{\partial L_b}{\partial x_{w,t,c}} \right|, \tag{1}$$

where $\mathcal{W}_s$ denotes the validation windows from subject s, and $b \in \{\mathrm{tot}, \mathrm{seq}, \mathrm{fing}, \mathrm{ord}\}$. Larger values therefore indicate greater local sensitivity of the corresponding task loss to perturbations of that sensor's input.

For visualization, gradient sensitivities were first averaged across seeds and folds for each subject and task. The resulting channel-wise vectors were then z-scored across sensors within each subject and task before projection onto the two-dimensional MEG sensor layout. This normalization emphasizes the relative spatial distribution of sensitivity rather than differences in overall gradient magnitude between subjects or tasks.

To quantify similarity between task-specific sensor profiles, we computed pairwise Pearson correlation and cosine similarity between the channel-wise gradient vectors for finger, sequence, and serial-position losses. Similarity was computed separately for each subject and subsequently summarized across subjects and random seeds.

B.9 SENSOR ABLATION

We assessed the contribution of anatomically grouped MEG sensors by replacing regional sensor time courses with matched Gaussian noise while keeping the trained model fixed. Analyses used the all-14 cohort and five-fold cross-validation checkpoints from the main experiments. For each checkpoint, we evaluated sequence, finger, and serial-position accuracy on the corresponding validation windows.

Sensors were divided into ten regions by hemisphere and lobe according to the CTF sensor layout map. For each region R and channel $c \in R$, the input was replaced during inference with independent Gaussian noise,

$$x_{t,c} \sim \mathcal{N}(\mu_c, \sigma_c^2),$$

where μ_c and σ_c were estimated from the corresponding validation data. All other channels and model parameters were unchanged.

For participant s, region R, and metric m, the ablation effect was defined as

$$\Delta_{s,R,m} = \mathrm{acc}_{s,R,m} - \mathrm{acc}_{s,\mathrm{baseline},m}.$$

Effects were averaged across seeds and folds before statistical testing.

B.10 STATISTICAL TESTS

Pairwise comparisons use two-sided Wilcoxon signed-rank tests on paired differences, with Pratt's treatment of zero differences (`zero_method=pratt`). For k-fold validation, accuracies are paired within seed across folds and the resulting fold-level differences are tested against the baseline.

For multiple comparisons within a metric panel, p-values are corrected using the Benjamini–Hochberg false discovery rate (FDR), with significance defined at $\alpha = 0.05$ on the adjusted p-values.

C HYPERPARAMETERS

Table 14: Configuration summary. These hyperparameter settings are used for Tables 1, Table 2.

Module	Substructure	Neuron2019	SpanishBCBL
Subject alignment	Hidden dim	384	256
	Dropout	0.0	0.75
Linear Channel mixer	Output dim	96	160
Multiscale conv.	Kernels	$\{3, 7, 10, 17, 32\}$	$\{2, 3, 5, 7, 16\}$
	Frequencies F	5	5
	Latents / frequency	3	5
First pool	Kernel	2	2
	Stride	2	2
Axial Transformer	Model dim	40	72
	Heads	4	4
	FF dim	192	224
	Layers	2	2
	Summary tokens	2	3
	Dropout	0.4	0.4
VAE MLP	Hidden dim	440	2040
	Hidden layers	1	0
	Dropout	0.1	0.1
Latent dim	Total	96	408
	Sequence / context	29	61
	Order	19	61
	Finger / key	48	286
Branch TCN	Kernels	$\{2, 5, 7\}$	$\{2, 5, 7\}$
	Dilations	$\{1, 2, 4\}$	$\{1, 2, 4\}$
	Strides	$\{2, 2, 2\}$	$\{2, 2, 2\}$
	Dropout	0.1	0.15
Condition dim	Sequence / sentence	178	1825
	Order	101	952
	Finger / key	304	1507
Training	Batch size	128	256
	Max windows / epoch	60k	all
	Max val windows / epoch	60k	all
	Learning rate	10^{-4}	1.01×10^{-4}
	Weight decay	10^{-2}	2.63×10^{-3}
	Epochs	50	50
	Early stopping	8	10
Loss weights	Context:Sequence / sentence	1.0	1.0
	Context:Word	—	1.0
	Target: Finger / key	3.0	5.0
	Event: Order	1.5	1.5
	KL schedule	$10^{-4} \to 5\times10^{-3}$	$10^{-4} \to 5\times10^{-3}$

Table 15: Hyperparameter settings for Table 2.

	HiLa (all-14)						BCBL					
Loss	$\lambda_{\mathbf{c}}$	$\lambda_{\mathbf{y}}$	$\lambda_{\mathbf{e}}$	Br.	$\lvert\mathbf{z}\rvert$	Latent dims	$\lambda_{\mathbf{c}}$	$\lambda_{\mathbf{y}}$	$\lambda_{\mathbf{e}}$	Br.	$\lvert\mathbf{z}\rvert$	Latent dims
$[\mathbf{y}]$	0.0	3.0	0.0	1	96	$(96, 0, 0)$	0.0	5.0	0.0	1	408	$(408, 0, 0)$
$[\mathbf{y} + \mathbf{e}]$	0.0	3.0	1.5	1	96	$(96, 0, 0)$	0.0	5.0	1.5	1	408	$(408, 0, 0)$
$[\mathbf{y} + \mathbf{c}]$	1.0	3.0	0.0	1	96	$(96, 0, 0)$	1.0	5.0	0.0	1	408	$(408, 0, 0)$
$[\mathbf{y} + \mathbf{c} + \mathbf{e}]$	1.0	3.0	1.5	1	96	$(96, 0, 0)$	1.0	5.0	1.5	1	408	$(408, 0, 0)$
$[\mathbf{y}]$, $[\mathbf{c}]$, $[\mathbf{e}]$	1.0	3.0	1.5	3	96	$(48, 29, 19)$	1.0	5.0	1.5	3	408	$(286, 61, 61)$

Table 16: Regularization settings for the adversarial and decorrelation ablations (Fig. S1). All settings use the same architecture and supervised loss weights as the main Neuron2019 configuration. Only the additional regularization term differs.

Setting	Independence penalty	Adversarial objective
Baseline	none	–
Decorrelation	correlation, $w = 0.5$	–
Group cross-covariance	group cross-cov., $w = 0.5$	–
All branches ← Participant	none	all latents ← Participant, $w = 0.5$
Sequence ← Finger	none	sequence latent ← finger, $w = 0.5$
Sequence ← Position	none	sequence latent ← position, $w = 0.5$
Sequence ← Participant	none	sequence latent ← Participant, $w = 0.5$
Finger ← Sequence	none	finger latent ← sequence, $w = 0.5$
Finger ← Position	none	finger latent ← position, $w = 0.5$
Finger ← Participant	none	finger latent ← Participant, $w = 0.5$
Position ← Sequence	none	position latent ← sequence, $w = 0.5$
Position ← Finger	none	position latent ← finger, $w = 0.5$
Position ← Participant	none	position latent ← Participant, $w = 0.5$