UNIBCI: TOWARDS A UNIFIED PRETRAINED MODEL FOR INVASIVE BRAIN-COMPUTER INTERFACES

Anonymous authors
Paper under double-blind review

ABSTRACT

Modeling invasive neural spike data is fundamental to advancing high-performance brain-computer interfaces (BCIs). However, existing approaches face critical challenges, including limited-scale heterogeneous data, cross-conditions distribution shift, and the intrinsic spatiotemporal complexity of invasive neural signals. In this work, we construct a pretraining corpus spanning multiple species, subjects, brain regions, and behavioral experiment paradigms. These heterogeneous recordings are standardized via our proposed unified normalization and tokenization. We also propose **UniBCI**, a **uni**fied pretrained model for **i**nvasive **B**rain-**C**omputer **I**nterfaces. The model integrates three key components: (1) a *context-conditioned spatio-temporal tokenization (CST)* scheme that embeds neural signals together with metadata into a shared representation space; (2) a hierarchical *Interval-Area Attention (IAA)* mechanism that captures patterns of spike dynamics in slots via linear attention and locality dependencies via sliding-window attention; and (3) a *self-supervised masked signals reconstruction* objective for rebuilding spike trains from unmasked neural tokens. We demonstrate that UniBCI outperforms baselines trained without behavioral data across both classification and regression tasks, while achieving performance comparable to leading behavior-supervised methods with fewer trainable parameters and lower inference latency. These results suggest that UniBCI provides a potential step toward general-purpose neural foundation models, enabling robust, scalable, and generalizable representation learning for invasive neural data. The code for this paper is available at: UniBCI Github.

1 INTRODUCTION

Invasive neural recording techniques, such as multi-electrode arrays (Tsai et al., 2017) and Neuropixels (Jun et al., 2017), provide high-resolution measurements of neuronal activity at the level of individual neurons and spikes, making them a critical substrate for studying neural computation and developing invasive brain-computer interfaces (iBCIs). However, invasive neural recordings are highly heterogeneous across datasets, with substantial variations in recording conditions, neural populations, brain regions, subjects, and experimental paradigms. Such heterogeneity poses a fundamental challenge to learning generalizable representations from neural spike data.

Early approaches to invasive neural data modeling primarily relied on classical statistical and dynamical-system methods, including Wiener filters (Glaser et al., 2020), Kalman filters (Wu et al., 2002), and Gaussian-Process Factor Analysis (GPFA) (Yu et al., 2008). While effective for capturing low-dimensional latent structures, these methods rely on strong assumptions of linearity and stationarity, restricting their capacity to model complex nonlinear neural dynamics. Deep learning approaches based on recurrent neural networks (RNNs) (Elman, 1990), Transformers (Vaswani et al., 2017), and state space models (SSMs) (Gu et al., 2022) provide more expressive modeling of neural dynamics and improve neural decoding performance. Methods such as NoMAD (Karpowicz et al., 2025) and CEBRA (Schneider et al., 2023) further address distribution shifts and representation alignment across recording conditions. Despite these advances, such approaches are generally designed for specific tasks or recording conditions and often require substantial task-specific data, hindering their ability to generalize across sessions, subjects, and experimental settings.

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.

More recently, large-scale pretraining has emerged as a promising paradigm for learning reusable neural representations across heterogeneous recording conditions. NDT1 (Ye & Pandarinath, 2021) introduced Transformer-based masked modeling for neural spike data, demonstrating the potential of pretraining to capture generalizable neural dynamics. NDT2 (Ye et al., 2023) further extended this framework to cross-session, cross-subject, and multi-context pretraining through a spatiotemporal Transformer architecture. However, the increased modeling capacity of NDT2 also incurs substantial computational and scaling costs. Multi-task Masking (MtM) (Zhang et al., 2024) explores self-supervised pretraining by learning multi-scale representations through masking and reconstruction across different spatial scales. NEDS (Zhang et al., 2025) extends multi-task masking to heterogeneous supervision signals by alternately masking neural, behavioral, and cross-modal modalities during training, enabling the model to jointly learn representations that are consistent across modalities. These approaches demonstrate the potential of self-supervised pretraining for improving the transferability of neural representations without relying exclusively on task-specific supervision.

In parallel, behavior-supervised pretraining has enabled neural foundation models to scale to substantially larger and more diverse datasets. NDT3 (Ye et al., 2025), for example, further scales neural decoding to hundreds of datasets by leveraging behavioral supervision. POYO (Azabou et al., 2023) and POSSM (Ryoo et al., 2025) introduce learnable neural token representations to facilitate scalable modeling across multiple subjects and recording sessions. Extending the tokenization framework of POYO, POYO+ (Azabou et al., 2025) incorporates task-specific query tokens together with session and task embeddings, enabling continuous regression, sequence classification, and segmentation within a shared cross-attention latent space. Despite these advances, existing methods have not fully addressed unified modeling of heterogeneous neural recordings across species, brain regions, and task paradigms, particularly without behavioral supervision.

We propose a unified neural pretrained modeling framework inspired by intrinsic characteristics of invasive brain signals. **First**, to address dataset fragmentation and massive data without behavioral labels, we construct a unified data framework with a multi-source pretraining corpus. We introduce a context-conditioned spatio-temporal tokenization (CST) strategy. The metadata consisting of species, subjects, brain regions, tasks, and experimental paradigms is encoded as context embeddings and injected into spike representations, enabling conditional modeling within a unified parameter space and improving robustness to domain shift. **Second**, Interval-Area Attention (IAA) is innovated to model neural dynamics at different levels of granularity. Within each interval, linear attention captures spiking patterns within a slot, and area-wise sliding-window attention models population dynamics among adjacent brain areas. This architecture enables joint modeling of temporal and spatial neural representations while improving computational efficiency. **Third**, we adopt a self-supervised masked spike reconstruction objective to learn general neural representations from unlabeled heterogeneous data. Our comprehensive experiments demonstrate performance with high efficiency on downstream datasets spanning multiple conditions.

2 Approach

We propose UniBCI, a unified spatio-temporal foundation model for neural spike data that integrates structured preprocessing, context-aware tokenization, and hierarchical attention layers. Spike trains from different datasets have varying shapes, where the dimension of timestep and channel is diverse. We suppress temporal information by binning timesteps and grouping channels to normalize multi-source spike trains (Section 2.1). Normalized spike trains combined with metadata contexts are chunked into tokens, and then the interval-area attention is employed in a stacked architecture. During pretraining, we utilize the self-supervised method of reconstructing masked tokens (Section 2.2).

2.1 Spike Train Normalization

Raw neural spike recordings are irregular in both temporal sampling and channel organization. We transform them into a structured tensor representation. Let $X_{\mathrm{raw}} \in \mathbb{R}^{T_{\mathrm{raw}} \times C_{\mathrm{raw}}}$ denote the raw spike matrix, where T_{raw} is the original temporal resolution and C_{raw} is the number of channels. Each element $X_{\tau,c}$ represents the spike activity of the channel c at time index τ. During the temporal

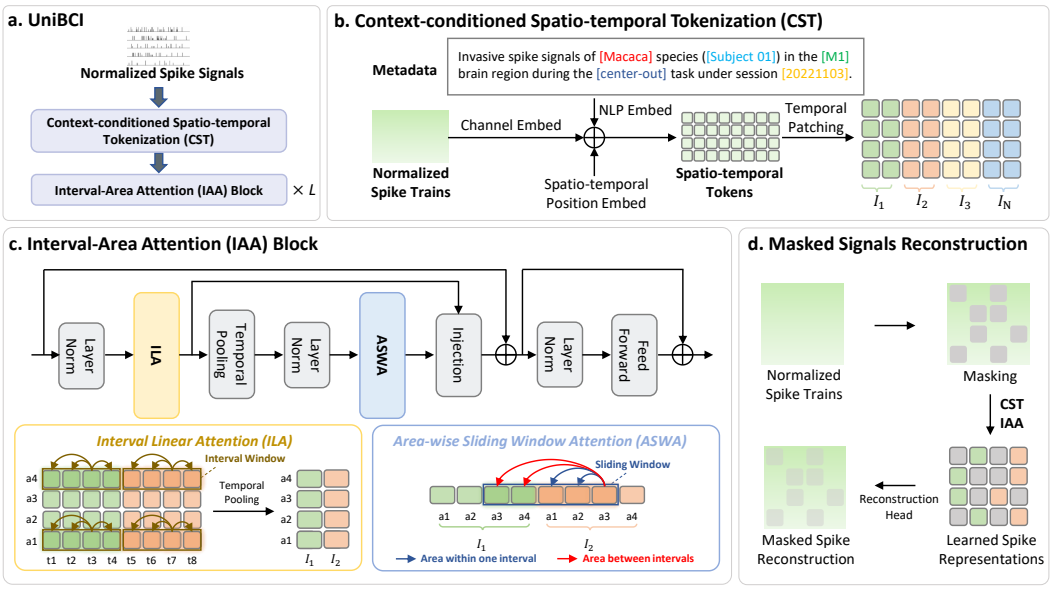

Figure 1: **a.** Architecture of UniBCI consists of context-aware tokenization and stacked interval-area attention layers. **b.** The normalized spike signal is tokenized with metadata. **c.** The interval embeddings pass through interval linear attention ILA to learn temporal features within a short slot, followed by area-wise sliding window attention ASWA to capture spatial dependencies among spikes. **d.** The reconstruction objective aims to generate masked parts of spike signals from extracted features of spike trains.

binning, we discretize time into bins and aggregate spikes:

$$\tilde{X}_{t,c} = \sum_{\tau \in \mathcal{B}(t)} X_{\tau,c}, \tag{1}$$

where $\mathcal{B}(t)$ is the set of timestamps in bin t. This maintains temporal information of spike activities, producing $\tilde{X} \in \mathbb{R}^{T_{\text{norm}} \times C_{\text{raw}}}$. Adopting the neural patching approach from NDT2 (Ye et al., 2023), we uniformly group the channels into A areas $\{C_a\}_{a=1}^{A}$, where each group contains an equal number of channels. The signals in a local area are denoted as

$$\hat{X}_{t,a} = \left[\tilde{X}_{t,c_1} \parallel \tilde{X}_{t,c_2} \parallel \cdots \parallel \tilde{X}_{t,c_{|C_a|}} \right], \tag{2}$$

where $|C_a|$ denotes the number of channels in the area a. We then truncate or pad channels in each area to a fixed number C_{norm}, resulting normalized spike trains $X_{\text{norm}} \in \mathbb{R}^{T_{\text{norm}} \times A \times C_{\text{norm}}}$.

2.2 UniBCI

Inspired by the characteristics of invasive neural activity, we propose UniBCI, which reconstructs masked spike tokens by optimizing the tokenization module and the interval-area attention blocks. This is illustrated in Figure 1. In the tokenization stage, normalized spike trains are chunked into spatio-temporal tokens by incorporating contextual representations. The interval and area-wise attention blocks that consider temporal and locality dependency are stacked to extract meaningful features of spike signals.

Context-conditioned Spatio-temporal Tokenization (CST): Given $X_{\text{norm}} \in \mathbb{R}^{T_{\text{norm}} \times A \times C_{\text{norm}}}$, we propose a tokenization strategy to serialize spike trains by integrating contexts and spatio-temporal positions. Specifically, a set of learnable weights $W_e \in \mathbb{R}^{C_{\text{norm}} \times d}$ and $b_e \in \mathbb{R}^d$ is shared among A areas, encoding channels into d-dimensional vectors:

$$X_{\text{emb}} = \Big\|_{a=1}^{A} (X_a \cdot W_e + b_e), \tag{3}$$

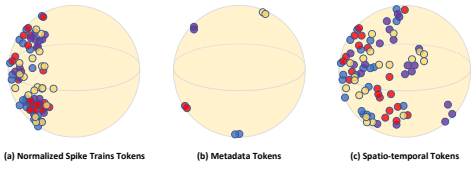

Figure 2: Context embeddings enrich the representation of spike tokens, ensuran ing expanded embedding space of spatio-temporal tokens.

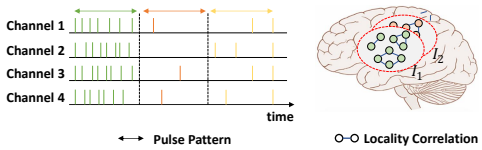

Figure 3: Spikes pulsing in a slot pattern differently and adjacent channels recorded in one area have locality correlation.

where $X_{\mathrm{a}} \in \mathbb{R}^{T_{\mathrm{norm}} \times C_{\mathrm{norm}}}$ denotes spike recordings of C_{norm} channels and T_{norm} timesteps in the area a, resulting in $X_{\mathrm{emb}} \in \mathbb{R}^{A \times T_{\mathrm{norm}} \times d}$ as embedded spike signals. Since spike trains stem from diverse paradigms, the metadata of species, subjects, brain regions, tasks and session identities are incorporated as the template:

Invasive spike signals of [Species] species ([Dataset] [Subject]) in the [Region] brain region during the [Task] task under session [Session/Date].

Leveraging a pretrained natural language model NLM, we embed contexts m in a dense space as vectors $X_{\mathrm{meta}} = \mathrm{NLM}(m) \in \mathbb{R}^d$. Furthermore, we employ two sets of learnable weights for embedding temporal and spatial positional embeddings, defined as $T_{\mathrm{pos}} \in \mathbb{R}^{1 \times T_{\mathrm{norm}} \times d}$ and $A_{\mathrm{pos}} \in \mathbb{R}^{A \times 1 \times d}$ respectively. With broadcast addition of the context embedding, the final $token \in \mathbb{R}^{T_{\mathrm{norm}} \times A \times d}$ is formulated as:

$$token = X_{\mathrm{emb}} \oplus X_{\mathrm{meta}} \oplus T_{\mathrm{pos}} \oplus A_{\mathrm{pos}}. \tag{4}$$

By combining the embedding of metadata, the hidden space of spikes is expanded, as illustrated in Figure 2. The semantic representations of metadata make the spatio-temporal tokens distinguishable from different sources. Eventually, we partition spike train tokens into N non-overlapping intervals over the temporal dimension, and the interval size is denoted as t :

$$I_i \in \mathbb{R}^{t \times A \times d}, \quad i = 1, \ldots, N. \tag{5}$$

Interval-Area Attention (IAA): Inspired by the characteristics of invasive signals, we design a stacked architecture where each layer considers interval and area dependency. As shown in Figure 3, the temporal patterning between local intervals is non-stationary (Brown et al., 2004), and the spatial grouping across channels is based on field correlation of neural populations (Buzsáki, 2004; Stevenson & Kording, 2011). Hence, we denote $H^\ell \in \mathbb{R}^{N \times A \times t \times d}$ as spike signals recorded from all areas A during all intervals N at layer ℓ. Since one interval of spikes is much shorter than whole span of brain activity, we choose the mechanism of linear attention (Katharopoulos et al., 2020) to accelerate inference speed although it doesn't reduce computational complexity in our case of $t < d$. We compute the representation of spikes $\tilde{O}^\ell_{i,a} \in \mathbb{R}^{t \times d}$ in one area from one interval $H^\ell_{i,a} \in \mathbb{R}^{t \times d}$:

$$\tilde{O}^\ell_{i,a} = H^\ell_{i,a} W^\ell_q \cdot ((H^\ell_{i,a} W^\ell_k)^\top \cdot H^\ell_{i,a} W^\ell_v), \tag{6}$$

where $W^\ell_q, W^\ell_k, W^\ell_v \in \mathbb{R}^{d \times d}$ are shared for N intervals and A areas. Concatenating embeddings of spikes generated from A areas during N intervals, the output of interval linear attention is denoted as $\mathrm{ILA}^\ell \in \mathbb{R}^{N \times A \times t \times d}$:

$$\mathrm{ILA}^\ell = \|_{i=1}^{N} \left(\|_{a=1}^{A} \tilde{O}^\ell_{i,a} \right). \tag{7}$$

Concerning the locality correlation, the method of sliding window attention (Beltagy et al., 2020; Zaheer et al., 2020b; Child et al., 2019) is employed. The queried area receives nearby signals and activities in the same area from historical intervals. Furthermore, this limits the attention scope in a sliding window, reducing quadratic computational complexity. We denote $\tilde{H}^\ell \in \mathbb{R}^{S \times d}$ as the input of the area-wise attention module where $S = N \times A$ after temporal pooling in interval size t and layer normalization:

$$\tilde{H}^\ell = \mathrm{LN}(\mathrm{AvgPool}(\mathrm{ILA}^\ell)). \tag{8}$$

We compute the area-wise sliding window attention written as $O^\ell \in \mathbb{R}^{S\times d}$:

$$O^\ell = \mathrm{softmax}(\tilde{H}^\ell W_Q^\ell \cdot (\tilde{H}^\ell W_K^\ell)^\top \odot M) \cdot \tilde{H}^\ell W_V^\ell. \tag{9}$$

where $M \in \mathbb{R}^{S\times S}$ is the sliding window mask where $M_{ij} = 1$ if $i - w + 1 \le j \le i$, otherwise $M_{ij} = -\infty$. Next, we reshape the output as $\mathrm{ASWA}^\ell = \mathrm{Reshape}(O^\ell) \in \mathbb{R}^{N\times A\times d}$. The output of the next layer is formulated as follows:

$$\mathrm{Z}^\ell = \mathrm{ILA}^\ell \oplus \mathrm{ASWA}^\ell + H^\ell, \tag{10}$$

$$H^{\ell+1} = \mathrm{FFN}(\mathrm{LN}(\mathrm{Z}^\ell)) + \mathrm{Z}^\ell. \tag{11}$$

The features extracted from the stacked layers are utilized to reconstruct the masked spike activity.

Masked Signals Reconstruction: Masking is performed directly on the normalized spike trains $X_{\mathrm{norm}} \in \mathbb{R}^{T_{\mathrm{norm}}\times A\times C_{\mathrm{norm}}}$, which are represented as a flattened sequence $\mathbf{X} \in \mathbb{R}^{J\times C_{\mathrm{norm}}}$ with total spatio-temporal length $J = T_{\mathrm{norm}} \times A$. We sample a binary mask $\mathcal{M} \in \{0,1\}^J$, where $\mathcal{M}_j = 1$ indicates that the j-th spatio-temporal activity vector is masked. Hence, the masked spike signals are formulated as:

$$\tilde{\mathrm{X}} = \{\tilde{\mathrm{x}}_j\}_{j=1}^J, \quad \tilde{\mathrm{x}}_j = \begin{cases} \mathrm{x}_j, & \mathcal{M}_j = 0, \\ 0, & \mathcal{M}_j = 1. \end{cases} \tag{12}$$

Followed by the context-conditioned spatio-temporal tokenization module and stacked interval-area attention blocks, the latent representation of masked spike signals is encoded as $\mathrm{H} = \mathrm{IAA}(\mathrm{CST}(\tilde{\mathrm{X}}))$. A lightweight reconstruction head $g_\phi(\cdot)$ maps the latent features back to the normalized spike activity space $\hat{\mathrm{X}} = g_\phi(\mathrm{H}) \in \mathbb{R}^{J\times C_{\mathrm{norm}}}$. The training objective is defined as a masked reconstruction loss, computed only over masked positions:

$$\mathcal{L}_{\mathrm{rec}} = \frac{\sum_{j=1}^J \mathcal{M}_j \cdot \ell\big(\hat{\mathrm{x}}_j, \mathrm{x}_j\big)}{\sum_{j=1}^J \mathcal{M}_j},$$

where $\ell(\cdot,\cdot)$ is typically instantiated as the mean squared error (MSE) of $\ell(\hat{\mathrm{x}}_j, \mathrm{x}_j) = \|\hat{\mathrm{x}}_j - \mathrm{x}_j\|_2^2$. This formulation enforces the model to infer missing spatio-temporal spike patterns from partially observed context.

Theoretical Justification: Our design is motivated by three complementary theoretical considerations. **First, context injection improves representation separability.** Neural spike embeddings are often low-dimensional due to the sparse and structured nature of population activity, whereas pretrained language representations provide a richer semantic basis. Their combination can increase the effective representation rank and separate samples from heterogeneous recording conditions, consistent with the benefits of language-guided representation learning observed in multimodal foundation models (Radford et al., 2021a).

Second, linear attention provides an appropriate inductive bias for local spike dynamics. Within short temporal intervals, neural activity is sparse and locally correlated. Linear attention replaces dense pairwise interactions with associative feature aggregation, reducing the dependence on sequence length while retaining the dominant temporal interactions (Katharopoulos et al., 2020). This is consistent with prior theoretical analyses showing that structured sparse attention can preserve expressive power while substantially improving scalability (Zaheer et al., 2020a).

Third, hierarchical interval-area attention promotes a shared latent representation. Neural population activity is widely characterized by low-dimensional latent dynamics shared across neurons and brain regions. By separately modeling local temporal dynamics and neighboring-area interactions, IAA performs structured aggregation that encourages representations from different intervals and areas to converge toward a common latent manifold.

3 Experiments

3.1 Datasets and Data Processing

We curated a diverse collection of datasets of neural spikes from non-human primates (NHPs), mice and humans to develop our model, as summarized in Table 1. The pretraining corpus consists of

Table 1: Pretraining and downstream datasets of invasive spike signals.

	Dataset	Species	Brain Regions	Tasks	# Indiv.	# Sess.	Data Scale
Pretraining	M1-CO1	Macaque	M1	CO	1	348	17.5 hours
	M1-CO2	Macaque	M1	CO	1	29	17.8 hours
	Pac-Man	Macaque	PMd, dlPFC, ACC	Pacman	1	121	109.5 hours
	HPC-HG	Macaque	HPC	Hidden Goal	1	36	44.8 hours
	LICK	Mouse	M2, SNR, VLS	Licking	11	104	74.1 hours
Downstream	M1-CO1 (New Sess.)	Macaque	M1	CO	1	24	1.3 hours
	LICK (New Indiv.)	Mouse	M2, SNR, VLS	Licking	3	15	11.4 hours
	Visual-Coding (Siegle et al., 2021)	Mouse	VISp	Drifting Gratings	17	17	45.6 hours
	PPC-FINGER (Guan et al., 2023)	Human	PPC	Finger Press	1	10	6.7 hours
	Perich (Perich et al., 2025)	Macaque	M1, PMd	CO, RT	1	12	2.5 hours
	MC-Maze (Pei et al., 2021)	Macaque	M1, PMd	Maze	1	1	1.9 hours
	Area2-Bump (Pei et al., 2021)	Macaque	S1	Bump	1	1	0.6 hours

large-scale recordings spanning multiple species, subjects, cortical and subcortical regions, and task paradigms. Specifically, we collect macaque datasets from M1-CO1, M1-CO2, Pac-Man and HPC-HG during center-out (CO) tasks, Pacman-game tracking and hidden goal navigation tasks. They encompass spike signals that span 189 hours covering brain regions of the primary motor cortex (M1), the premotor cortex (PMd), the dorsolateral prefrontal cortex (dlPFC), the anterior cingulate cortex (ACC) and the hippocampus (HPC). The pretraining corpus is further complemented by the LICK mouse dataset, which covers secondary motor cortex (M2), substantia nigra pars reticulata (SNR), and ventrolateral striatum (VLS) during licking behavior across 104 sessions. Downstream benchmarks contain both classification and regression tasks. The classification benchmarks include the M1-CO1, LICK, Visual-Coding (Siegle et al., 2021), and PPC-FINGER (Guan et al., 2023) datasets, which involve center-out reaching, multi-session licking behavior, mouse visual motion direction decoding, and human finger press movement decoding, respectively. We assessed the model's generalization ability across different recording conditions and species. The regression benchmarks include the Perich (Perich et al., 2025) dataset and the Neural Latent Benchmark suite (Pei et al., 2021). Specifically, we evaluated cross-subject transfer on the Perich dataset in both center-out and random target (RT) reaching tasks. The MC-Maze (Pei et al., 2021) task investigates autonomous motor dynamics during continuous movement generation, while the Area2-Bump (Pei et al., 2021) task recording responses to externally driven sensory perturbations in the somatosensory cortex (S1).

All datasets underwent standardized preprocessing, including spike binning, temporal alignment with behavioral kinematics (e.g., hand velocity or 2D coordinates), and sample segmentation via sliding windows or trial-based adaptive binning.

3.2 Experimental Results

We compare the proposed method with several baselines on both classification and regression tasks using a comprehensive set of metrics. For models trained from scratch, we include Wiener filters (WF) (Glaser et al., 2020), a gated recurrent unit (GRU) (Cho et al., 2014), a multilayer perceptron (MLP), and NoMAD (Karpowicz et al., 2025). For pretrained models, we compare against a variational autoencoder (VAE) (Kingma & Welling, 2013), NDT1 (Ye & Pandarinath, 2021), NDT2 (Ye et al., 2023), MtM (Zhang et al., 2024), POYO (Azabou et al., 2023), and POYO+ (Azabou et al., 2025). All pretrained baseline models are trained on the same dataset as UniBCI for a fair comparison, with POYO and POYO+ being the only baselines that additionally require behavioral labels during pretraining. For classification tasks, we report Balanced Accuracy and Weighted F1 score while for regression tasks, we utilize the coefficient of determination (R^2).

For downstream evaluation, the pretrained model is fine-tuned together with task-specific heads under a full-parameter training setting. The fine-tuning process uses a learning rate of 1×10^{-4} and a batch size of 64. We utilized 80% of each downstream dataset for training and reserved the remaining 20% for testing. All downstream results are reported as the mean $\pm$ standard deviation over five random seeds, ranging from 42 to 46. For M1-CO1, we adopt two settings: (1) Multi-day, where sessions from each day are split for joint training and testing; (2) Cross-day, where models are trained on a subset of days and tested on unseen days. For LICK, we train and test separately

Table 2: Performance comparison on classification and regression tasks. Values represent mean ± standard deviation where applicable. The *Generalization* column indicates the transfer setting. **Bold** and underlined values indicate the best and second-best performance, respectively. ⋆ indicates models pretrained with behavioral labels.

			From scratch				Pretrained						
Generalization	Dataset	Metrics	WF	GRU	MLP	NoMAD	VAE	NDT1	NDT2	MtM	POYO⋆	POYO+⋆	UniBCI (Ours)
Classification Tasks													
New Session	M1-CO1 (Multi-day)	B-Acc	0.645±0.008	0.806±0.005	0.680±0.006	0.848±0.004	0.817±0.005	0.885±0.006	0.863±0.003	0.854±0.007	0.859±0.003	0.892±0.007	**0.900±0.008**
		F1-W	0.649±0.007	0.808±0.005	0.680±0.006	0.850±0.004	0.820±0.005	0.886±0.006	0.866±0.002	0.855±0.006	0.858±0.003	0.893±0.006	**0.901±0.008**
	M1-CO1 (Cross-day)	B-Acc	0.623±0.012	0.739±0.010	0.666±0.008	0.821±0.006	0.758±0.009	0.830±0.019	0.828±0.013	0.843±0.001	**0.870±0.007**	0.860±0.016	0.855±0.014
		F1-W	0.624±0.012	0.741±0.009	0.660±0.008	0.822±0.006	0.756±0.009	0.828±0.019	0.828±0.013	0.845±0.012	**0.870±0.007**	0.859±0.016	0.855±0.014
New Subject	LICK	B-Acc	0.655±0.005	0.705±0.004	0.662±0.003	0.724±0.003	0.685±0.004	0.739±0.001	0.745±0.001	0.741±0.002	0.729±0.007	**0.777±0.002**	0.759±0.003
		F1-W	0.792±0.004	0.781±0.003	0.851±0.002	0.855±0.002	0.843±0.002	0.865±0.000	0.865±0.001	0.865±0.001	0.854±0.001	**0.866±0.000**	0.859±0.002
New Specie	PPC-FINGER	B-Acc	0.852±0.006	0.894±0.005	0.929±0.004	0.717±0.006	0.881±0.005	0.962±0.004	0.965±0.001	0.950±0.003	0.835±0.007	0.870±0.007	**0.968±0.003**
		F1-W	0.824±0.006	0.894±0.005	0.927±0.004	0.719±0.006	0.884±0.005	0.960±0.003	0.963±0.002	0.951±0.003	0.832±0.008	0.870±0.006	**0.967±0.003**
New Task	Visual-Coding	B-Acc	0.737±0.004	0.686±0.006	0.754±0.007	0.734±0.005	0.698±0.004	0.750±0.004	0.759±0.002	0.773±0.005	0.680±0.004	0.702±0.008	**0.782±0.004**
		F1-W	0.729±0.003	0.677±0.005	0.751±0.007	0.728±0.005	0.670±0.003	0.747±0.004	0.757±0.001	0.773±0.005	0.678±0.005	0.702±0.007	**0.784±0.004**
		Avg. B-Acc	0.702	0.766	0.738	0.769	0.768	0.833	0.832	0.832	0.795	0.821	**0.853**
Regression Tasks													
New Subject	Perich T-CO	R^2	0.580±0.008	0.625±0.006	0.682±0.005	0.656±0.006	0.716±0.004	0.747±0.002	0.706±0.004	0.769±0.002	0.756±0.005	0.731±0.007	**0.770±0.004**
New Task	Perich T-RT	R^2	0.559±0.009	0.621±0.007	0.613±0.007	0.605±0.008	0.629±0.007	0.681±0.003	0.633±0.002	0.673±0.002	**0.693±0.003**	0.684±0.006	0.693±0.004
	MC-Maze	R^2	0.728±0.006	0.773±0.005	0.804±0.004	0.795±0.004	0.797±0.004	0.884±0.002	0.877±0.001	0.868±0.004	0.880±0.005	0.879±0.004	**0.886±0.003**
New Region	Area2-Bump	R^2	0.724±0.006	0.814±0.004	0.778±0.005	0.793±0.004	0.797±0.004	**0.891±0.003**	0.878±0.003	0.855±0.007	0.862±0.004	0.875±0.003	0.888±0.002
		Avg. R^2	0.648	0.708	0.719	0.712	0.735	0.801	0.774	0.791	0.798	0.792	**0.809**

on each of the three individuals, and report the average performance across subjects. For all other datasets, evaluation is conducted within each session, with results averaged across sessions when applicable.

We evaluate UniBCI across a wide range of downstream settings, with results reported in Table 2. Under the classification task, UniBCI achieves state-of-the-art performance in new session settings for M1-CO1 (Multi-day), reaching a balanced accuracy of 0.900, and maintains competitive performance with a balanced accuracy of 0.855 on M1-CO1 (Cross-day). On new subject evaluation for LICK, it achieves a strong balanced accuracy of 0.759, performing on par with leading baselines. On new species, it achieves the best balanced accuracy of 0.968 on the human PPC-FINGER, validating its cross-species transfer capability. For regression tasks, UniBCI yields an overall best average R^2 of 0.809 across Perich, MC-Maze, and Area2-Bump datasets, consistently surpassing strong pretrained baselines of NDT1(0.801) and POYO(0.798).

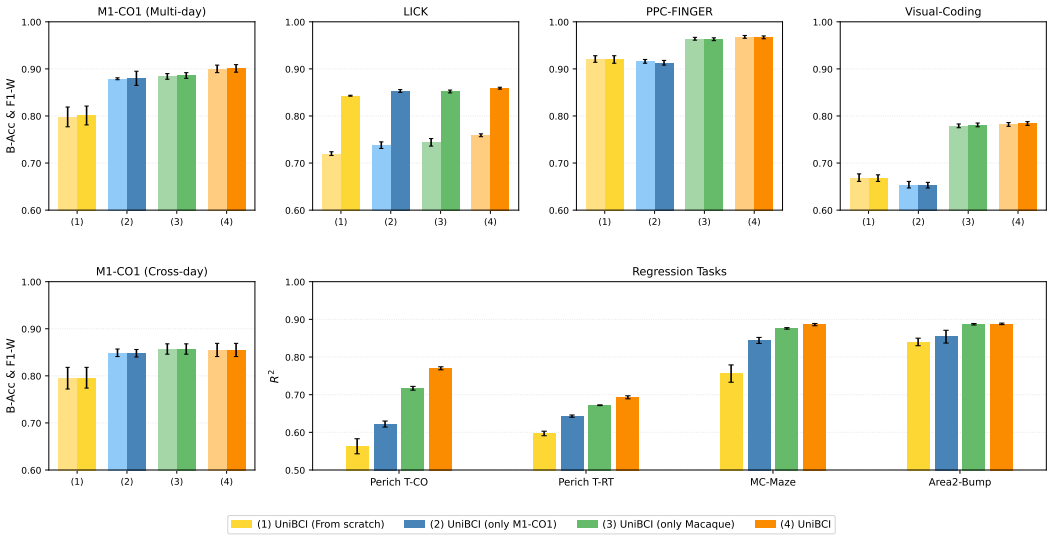

Figure 4: Ablation study of pretraining strategies across various classification and regression datasets. Each group in the classification task displays the Balanced Accuracy (light colors) and Weighted F1-score (dark colors) for different UniBCI variants.

3.3 Effectiveness of Pretraining Corpus

As shown in Fig. 4, we evaluate how pretraining strategies and data diversity affect UniBCI's generalization across four settings: (1) training from scratch, (2) pretraining on M1-CO1 only, (3) multi-task pretraining on macaque, and (4) full cross-species pretraining. The from-scratch baseline (1) suffers substantial performance degradation. Without self-supervised pretraining on diverse neural signals, metadata embeddings of species, brain regions, and tasks remain poorly aligned with neural dynamics, limiting text-conditioned representation learning. Pretraining on M1-CO1 alone (2) improves macaque motor tasks; by contrast, this setting shows no generalization to the unseen domain of human (PPC-FINGER) and even degrades the performance of the Visual-Coding task. Interestingly, the LICK task is beneficial when one center-out task is pretrained. M1-CO1 (macaque M1) and LICK (mouse M2) both involve motor-related cortical regions, suggesting that our unified model can effectively transfer motor-dynamics priors across species. However, these motor-centric priors may be less transferable to sensory (VISp) and higher-order cognitive (PPC) domains. Expanding the corpus with diverse macaque regions and tasks in (3) broadens the functional and semantic coverage of the pretraining corpus, uniformly enhancing downstream performance, particularly on PPC-FINGER and Visual-Coding. Finally, adding rodent data in (4) further improves most downstream tasks. This suggests that cross-species metadata enriches the text-conditioned representation space, helping disentangle species-specific semantics from shared neural dynamics and strengthening species-invariant text-to-neural mappings. Overall, these results indicate that increasing the semantic and functional diversity of pretraining metadata, rather than data volume alone, is key to fully exploiting the shared text-neural space and capturing generalizable neural coding principles across species.

3.4 Structural Ablation Study

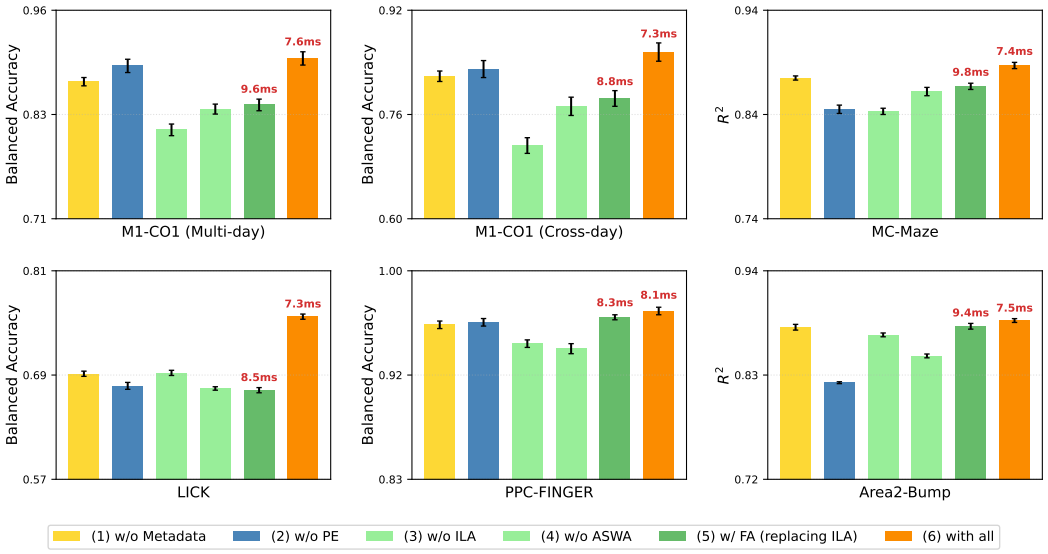

Figure 5: The experimental results of ablating modules. The red values in the subplot indicate the inference latency of Varient (5) and (6).

We conduct ablation studies to evaluate the contribution of each component, as shown in Figure 5. Removing metadata (Variant 1) leads to consistent performance drops across all datasets, with a 2.8% decrease in balanced accuracy on M1-CO1 (Multi-day) and a 1.2% drop on MC-Maze, indicating that contextual information provides useful priors and helps mitigate domain shifts. Eliminating positional encoding (Variant 2) results in performance drops by 4.2% and 6.5% on MC-Maze and Area2-Bump, respectively. This highlights the importance of explicit spatial and temporal order in modeling neural dynamics. Variants (3) and (4) evaluate the interval-area attention mechanism. Removing ILA causes a 8.6% drop on M1-CO1 (Multi-day), demonstrating its role in capturing fine-grained temporal patterns. Without the module of ASWA leads to consistent declines across

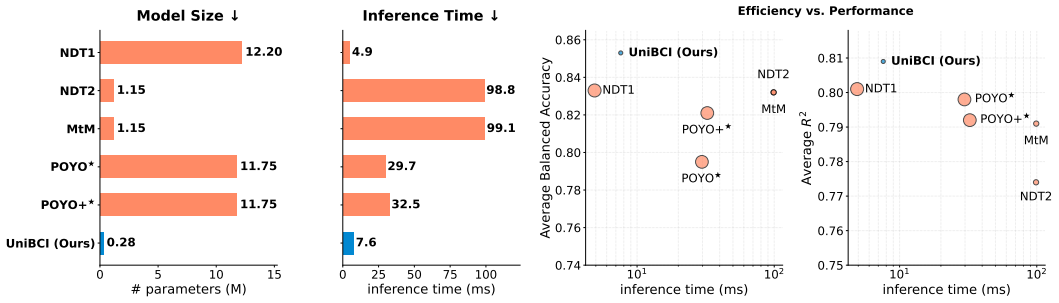

Figure 6: Comparison of pretrained model efficiency in terms of parameter count, inference latency and balanced performance. The x-axis is in log scale to better visualize the performance of low-latency models, and circle size represents the parameter count. $\star$ indicates models pretrained with behavioral labels.

datasets, including 6.1% on M1-CO1 (Multi-day) and 3.7% on Area2-Bump, underscoring the importance of modeling global spatial dependencies. Furthermore, replacing ILA mechanism with full attention (Variant 5) causes consistent performance drops and increases latency, proving that ILA offers a more effective and efficient inductive bias for neural dependencies. The full model achieves the best performance across all benchmarks, confirming the complementary benefits of all components. These results show that metadata embeddings and hierarchical interval-area attention jointly enable effective multi-scale spatiotemporal representation learning.

3.5 Less Parameters with High Efficiency

To evaluate computational efficiency, we report the number of trainable parameters of the model backbone and inference time per sample in Figure 6. UniBCI incorporates a pretrained text encoder with 22.7M parameters, which is frozen during training and thus introduces no additional training cost. As shown in Figure 6, UniBCI achieves strong model compactness, with only 0.28M trainable parameters, approximately 42 times smaller than NDT1 and POYO. To evaluate real-time deployment capability, we establish a standardized inference latency protocol. All models were evaluated on a single NVIDIA A100 GPU using single-precision (FP32) arithmetic with a batch size of 1 to simulate online streaming. Following 50 GPU warm-up passes, inference times were averaged over 16 samples per dataset across all five benchmarks (M1-CO1, LICK, MC-Maze, PPC-FINGER, and Area2-Bump). Under this protocol, NDT1 achieves the lowest latency (4.9 ms) but suffers from degraded accuracy and R^2 performance. In contrast, UniBCI maintains a competitive inference time of 7.6 ms while consistently outperforming all baselines, achieving both the highest average R^2 of 0.809 and balanced accuracy of 0.853. These results show that UniBCI's inference time is well within the optimal real-time BCI decoding latency threshold ($\leq$ 10 ms), demonstrating its strong potential for real-time BCI decoding applications.

4 Conclusion

We present UniBCI, a unified neural foundation model for invasive spike data that addresses key challenges in neural decoding, including data heterogeneity with limited scale, cross-domain distribution shift and spatiotemporal complexity of invasive signals. By integrating context-conditioned tokenization, the proposed framework enables scalable and generalizable representation learning across diverse neural datasets. We introduce a hierarchical interval-area attention module to capture fine-grained features of spike signals across intervals and areas. In pretraining, self-supervised masked reconstruction requires no behavioral labels, thereby optimizing UniBCI efficiently. Comprehensive experiments demonstrate that UniBCI consistently outperforms existing methods on both classification and regression benchmarks, suggesting strong generalization across species, subjects, sessions and task paradigms. Moreover, the model maintains high efficiency with a compact architecture, making it suitable for real-time BCI applications.

AI USE STATEMENT

In this work, we used generative AI tools to assist with code development, edit the manuscript for readability, and improve the clarity of its language. We did not use generative AI tools for any of the research tasks requiring disclosure under the ICLR AI policy. All AI-assisted work was reviewed and verified by the authors, who take full responsibility for the final content of this work.

REFERENCES

Mehdi Azabou, Vinam Arora, Venkataramana Ganesh, Ximeng Mao, Santosh Nachimuthu, Michael Mendelson, Blake Richards, Matthew Perich, Guillaume Lajoie, and Eva Dyer. A unified, scalable framework for neural population decoding. *Advances in Neural Information Processing Systems*, 36:44937–44956, 2023.

Mehdi Azabou, Krystal Xuejing Pan, Vinam Arora, Ian Jarratt Knight, Eva L Dyer, and Blake Aaron Richards. Multi-session, multi-task neural decoding from distinct cell-types and brain regions. In *The Thirteenth International Conference on Learning Representations*, 2025.

Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Emery N Brown, Robert E Kass, and Partha P Mitra. Multiple neural spike train data analysis: state-of-the-art and future challenges. *Nature neuroscience*, 7(5):456–461, 2004.

György Buzsáki. Large-scale recording of neuronal ensembles. *Nature neuroscience*, 7(5):446–451, 2004.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Kyunghyun Cho, Bart Van Merriënboer, Çağlar Gulçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1724–1734, 2014.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

Joshua I Glaser, Ari S Benjamin, Raeed H Chowdhury, Matthew G Perich, Lee E Miller, and Konrad P Kording. Machine learning for neural decoding. *eneuro*, 7(4), 2020.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *The International Conference on Learning Representations (ICLR)*, 2022.

Charles Guan, Tyson Aflalo, Kelly Kadlec, Jorge Gámez de Leon, Emily R Rosario, Ausaf Bari, Nader Pouratian, and Richard A Andersen. Decoding and geometry of ten finger movements in human posterior parietal cortex and motor cortex. *Journal of Neural Engineering*, 20(3):036020, 2023.

Binjie Hong, Zihang Xu, Tengyu Zhang, and Tielin Zhang. Bidirectional cross-day alignment of neural spikes and behavior using a hybrid snn-ann algorithm. *Frontiers in Neuroscience*, 20: 1772958, 2026.

James J Jun, Nicholas A Steinmetz, Joshua H Siegle, Daniel J Denman, Marius Bauza, Brian Barbarits, Albert K Lee, Costas A Anastassiou, Alexandru Andrei, Çağatay Aydın, et al. Fully integrated silicon probes for high-density recording of neural activity. *Nature*, 551(7679):232–236, 2017.

Brianna M. Karpowicz, Yahia H. Ali, Lahiru N. Wimalasena, Andrew R. Sedler, Mohammad Reza Keshtkaran, Kevin Bodkin, Xuan Ma, Daniel B. Rubin, Ziv M. Williams, Sydney S. Cash, Leigh R. Hochberg, Lee E. Miller, and Chethan Pandarinath. Stabilizing brain-computer interfaces through alignment of latent dynamics. *Nature Communications*, 16(1):4662, 2025. doi: 10.1038/s41467-025-59652-y.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 5156–5165. PMLR, 2020.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 2002.

Xuan Ma, Fabio Rizzoglio, Kevin L Bodkin, Eric Perreault, Lee E Miller, and Ann Kennedy. Using adversarial networks to extend brain computer interface decoding accuracy over time. *elife*, 12: e84296, 2023.

Georgios Mentzelopoulos, Ioannis Asmanis, Konrad Kording, Eva L Dyer, Kostas Daniilidis, and Flavia Vitale. A scalable, causal, and energy efficient framework for neural decoding with spiking neural networks. *Advances in Neural Information Processing Systems*, 38:49705–49747, 2026.

Felix Pei, Joel Ye, David M. Zoltowski, Anqi Wu, Raeed H. Chowdhury, Hansem Sohn, Joseph E. O'Doherty, Krishna V. Shenoy, Matthew T. Kaufman, Mark Churchland, Mehrdad Jazayeri, Lee E. Miller, Jonathan Pillow, Il Memming Park, Eva L. Dyer, and Chethan Pandarinath. Neural latents benchmark '21: Evaluating latent variable models of neural population activity. In *Advances in Neural Information Processing Systems (NeurIPS), Track on Datasets and Benchmarks*, 2021.

Matthew G Perich, Lee E Miller, Mehdi Azabou, and Eva L Dyer. Long-term recordings of motor and premotor cortical spiking activity during reaching in monkeys. *Data set*, 2025.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 8748–8763. PMLR, 2021a.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021b.

Avery Hee-Woon Ryoo, Nanda H Krishna, Ximeng Mao, Mehdi Azabou, Eva L Dyer, Matthew G Perich, and Guillaume Lajoie. Generalizable, real-time neural decoding with hybrid state-space models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

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

Joshua H Siegle, Xiaoxuan Jia, Séverine Durand, Sam Gale, Corbett Bennett, Nile Graddis, Greggory Heller, Tamina K Ramirez, Hannah Choi, Jennifer A Luviano, et al. Survey of spiking in the mouse visual system reveals functional hierarchy. *Nature*, 592(7852):86–92, 2021.

Yue Song, T. Anderson Keller, Yisong Yue, Pietro Perona, and Max Welling. Langevin flows for modeling neural latent dynamics. 2025.

Ian H Stevenson and Konrad P Kording. How advances in neural recording affect data analysis. *Nature neuroscience*, 14(2):139–142, 2011.

David Tsai, Daniel Sawyer, Adrian Bradd, Rafael Yuste, and Kenneth L Shepard. A very large-scale microelectrode array for cellular-resolution electrophysiology. *Nature Communications*, 8 (1):1802, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Jiquan Wang, Sha Zhao, Zhiling Luo, Yangxuan Zhou, Haiteng Jiang, Shijian Li, Tao Li, and Gang Pan. CBramod: A criss-cross brain foundation model for EEG decoding. In *The Thirteenth International Conference on Learning Representations*, 2025.

Wenhui Wang, Hangbo Bao, Shaohan Huang, Li Dong, and Furu Wei. MiniLMv2: Multi-head self-attention relation distillation for compressing pretrained transformers. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 2140–2151, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.188.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

Wei Wu, M Black, Yun Gao, M Serruya, A Shaikhouni, J Donoghue, and Elie Bienenstock. Neural decoding of cursor motion using a kalman filter. *Advances in neural information processing systems*, 15, 2002.

Yuqi Yang, Tengjun Liu, Haiyan Zhang, Wang Ruixue, Xuchao Chen, Mingkang Li, Yansong Chua, Nenggan Zheng, and Shaomin Zhang. Pretraining with re-parametrized self-attention: Unlocking generalizationin snn-based neural decoding across time, brains, and tasks. In *International Conference on Learning Representations*, volume 2026, pp. 19693–19722, 2026.

Joel Ye and Chethan Pandarinath. Representation learning for neural population activity with Neural Data Transformers. *Neurons, Behavior, Data analysis, and Theory*, August 2021. ISSN 2690-2664. doi: 10.51628/001c.27358.

Joel Ye, Jennifer Collinger, Leila Wehbe, and Robert Gaunt. Neural data transformer 2: multi-context pretraining for neural spiking activity. *Advances in Neural Information Processing Systems*, 36:80352–80374, 2023.

Joel Ye, Fabio Rizzoglio, Adam Smoulder, Hongwei Mao, Xuan Ma, Patrick Marino, Raeed Chowdhury, Dalton Moore, Gary Blumenthal, William Hockeimer, et al. A generalist intracortical motor decoder. *bioRxiv*, 2025.

Byron M Yu, John P Cunningham, Gopal Santhanam, Stephen Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *Advances in neural information processing systems*, 21, 2008.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big bird: Transformers for longer sequences. In *Advances in Neural Information Processing Systems*, volume 33, 2020a.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297, 2020b.

Yizi Zhang, Yanchen Wang, Donato M Jiménez-Benetó, Zixuan Wang, Mehdi Azabou, Blake Richards, Renee Tung, Olivier Winter, Eva Dyer, Liam Paninski, et al. Towards a" universal translator" for neural dynamics at single-cell, single-spike resolution. *Advances in Neural Information Processing Systems*, 37:80495–80521, 2024.

Yizi Zhang, Yanchen Wang, Mehdi Azabou, Alexandre Andre, Zixuan Wang, Hanrui Lyu, The International Brain Laboratory, Eva Dyer, Liam Paninski, and Cole Hurwitz. Neural encoding and decoding at scale. *arXiv preprint arXiv:2504.08201*, 2025.

Yuchen Zhou, Jiamin Wu, Zichen Ren, Zhouheng Yao, Weiheng Lu, Kunyu Peng, Qihao Zheng, Chunfeng Song, Wanli Ouyang, and Chao Gou. Csbrain: A cross-scale spatiotemporal brain foundation model for eeg decoding. *arXiv preprint arXiv:2506.23075*, 2025.

APPENDIX

A RELATED WORK

A.1 NEURAL DECODING METHODS

Neural decoding aims to infer behavioral or cognitive variables from neural activity. Early approaches were primarily based on linear statistical models and dynamical systems, such as Kalman filters (Wu et al., 2002), Wiener filters (Glaser et al., 2020), and generalized linear models (GLMs). While effective at capturing low-dimensional latent structures, these methods rely on linearity and stationarity assumptions, limiting their ability to model complex nonlinear neural dynamics. With the rise of deep learning, architectures including recurrent neural networks (RNNs) (Elman, 1990), convolutional neural networks (CNNs) (LeCun et al., 2002), Transformers (Vaswani et al., 2017), and state space models (SSMs) (Gu et al., 2022) have been widely adopted for neural signal modeling, improving performance in tasks such as motor decoding and behavior prediction. These models better capture nonlinear temporal dependencies, but are typically restricted to single-session or single-task settings, and still suffer from limited generalization across sessions, subjects, and experimental conditions. To improve cross-session stability, alignment-based methods have been proposed. For instance, CycleGAN (Ma et al., 2023) and NoMAD (Karpowicz et al., 2025) train a decoder on a reference session (e.g., day one) and subsequently learn mappings to align neural activity from later sessions. AlignNet (Hong et al., 2026) further integrates spiking neural networks (SNNs), artificial neural networks (ANNs), and contrastive learning to align neural signals and behavioral semantics into a shared latent space, enabling robust cross-day bidirectional decoding and simulation. Despite their effectiveness under controlled settings, these methods still exhibit notable generalization gaps when applied to unseen subjects or tasks. Meanwhile, efficiency has emerged as another important direction in neural decoding. Spikachu (Mentzelopoulos et al., 2026) explores compact spiking neural networks (SNNs) for continuous and causal neural decoding with low energy consumption. Furthermore, RAT-SNN (Yang et al., 2026) introduces re-parameterized self-attention into SNN-based pretraining, enabling generalization across brains and tasks within spiking architectures. While these methods emphasize energy and computational efficiency through sparse spiking dynamics, UniBCI pursues a complementary direction by combining metadata-conditioned representation learning with a highly compact trainable backbone, achieving flexible cross-species and cross-task generalization with only 0.28M trainable parameters.

A.2 FOUNDATION MODELS FOR INVASIVE SIGNALS

The emergence of large-scale self-supervised foundation models is reshaping machine learning, demonstrating strong cross-task transferability in domains such as natural language processing (Radford et al., 2018; 2019; Wei et al., 2021) and computer vision (Radford et al., 2021b; Kirillov et al., 2023). This paradigm is increasingly being extended to neuroscience. In neural decoding, recent works explore large-scale pretraining strategies. On the self-supervised side, NDT1 (Ye & Pandarinath, 2021) introduces masked modeling for neural spike sequence reconstruction using a Transformer, enabling parallel (non-recurrent) temporal modeling via masked time segments. NDT2 (Ye et al., 2023) extends this framework to a spatiotemporal Transformer and improves generalization through cross-session and cross-subject pretraining. MtM (Zhang et al., 2024) further proposes

multi-level masking strategies (across time, neurons, and brain regions) to learn more general neural representations. Under behavior-supervised settings, POYO (Azabou et al., 2023) leverages Perceiver IO–based cross-attention to tokenize spike events, enabling large-scale multi-session training without explicit neuron alignment; its extension, POYO+ (Azabou et al., 2025), enhances adaptability across cell types and brain regions. NDT3 (Ye et al., 2025) scales autoregressive Transformers to hundreds of heterogeneous datasets, achieving strong zero-shot and few-shot generalization. POSSM (Ryoo et al., 2025) combines a cross-attention encoder with a recurrent SSM backbone, improving inference efficiency while maintaining competitive performance. NEDS (Zhang et al., 2025) introduces a multi-task masking strategy, in which masking is alternated across neural, behavioral, and cross-modal modalities during training, enabling the model to jointly learn representations that are consistent across heterogeneous supervision signals. Nevertheless, existing approaches have only recently begun to explore broader cross-species and cross-domain generalization, while jointly achieving robust transfer across brain regions, species, and tasks remains challenging.

A.3 Spatiotemporal Modeling of Neural Dynamics

Neural signals exhibit rich multi-scale spatiotemporal dependencies, making their effective modeling a central challenge in neural decoding. Traditional approaches, such as RNNs (Elman, 1990) and SSMs (Gu et al., 2022), are capable of modeling temporal dependencies but often struggle with long-range interactions. Transformers, with self-attention mechanisms, enable direct modeling of dependencies across arbitrary time steps, substantially improving long-sequence modeling. In non-invasive neural signals (e.g., EEG and MEG), Transformer-based spatiotemporal models have been widely explored. For example, CBraMod (Wang et al., 2025) introduces a criss-cross attention mechanism that decouples temporal and spatial modeling to address signal heterogeneity, while CSBrain (Zhou et al., 2025) proposes a cross-scale spatiotemporal framework with hierarchical tokenization and structured sparse attention to capture features ranging from local activations to slow oscillatory dynamics. For invasive neural signals, NDT2 (Ye et al., 2023) adopts a ViT-style (Dosovitskiy et al., 2020) spatiotemporal patch tokenization, where recorded neurons (channels) are spatially arranged and grouped into tokens for unified modeling of neural dynamics. However, this design incurs substantial computational overhead. Building upon this paradigm, our tokenization strategy uniformly groups neurons and further partitions them into local blocks. We apply interval linear attention alongside area sliding window attention, reducing computational complexity while preserving the ability to capture both local and global spatiotemporal dependencies.

B Model Details and Experimental Settings

B.1 Model Details and Pretraining Settings

B.1.1 Model Implementation.

To achieve cross-dataset alignment and generalizable neural representation learning, we design a transformer-based pretraining model, with detailed configurations reported in Table 3. During spike train normalization, the random channels are divided into 8 areas with the area size of 32, allowing the model to adapt to different electrode layouts or neural population structures. In the tokenization stage, we leverage a pretrained language model, MiniLM-L6-v2 (Wang et al., 2021), to encode textual metadata. The encoded textual features are projected into the same embedding space as neural signals via a linear projection layer. Next, the spatio-temporal tokens are partitioned into 10 intervals, each with 10 timesteps. We compose 4 interval-area attention blocks, each equipped with 8 heads and an embedding dimension of 64 for both interval linear attention (ILA) and area sliding window attention (ASWA) modules. We further set a window size of 10 in ASWA to model area-wise dependencies efficiently while maintaining computational tractability.

B.1.2 Pretraining Strategy.

In pretraining, the masking ratio is set to 50%. The model reconstructs the masked segments based on the visible tokens without behavioral labels. The training objective is defined as the mean squared error (MSE) between the reconstructed and ground-truth signals. We employ the AdamW optimizer with an initial learning rate of 5×10^{-4} and apply a cosine annealing schedule for learning rate

decay. The model is trained for 40 epochs with a batch size of 128. All experiments are conducted on 8 NVIDIA A100 GPUs.

Table 3: Model architecture and pretraining hyperparameters.

Category	Hyperparameters	Settings
Architecture	Embedding Dimension	64
	Number of Layers	4
	Attention Heads	8
	Sliding Attention Window Size	10
Tokenization	Interval Size	10
	Area Size	32
	Area Number	8
Language	Text Encoder	MiniLM-L6-v2
	Text Encoding	Enabled
Pre-training	Epochs	40
	Batch Size	128
	Masking Ratio	0.5
	Dropout	0.1
	Optimizer	AdamW
	Learning Rate	5e-4
	Weight Decay	5e-2
	Scheduler	CosineAnnealingLR
	Minimal Learning Rate	1e-5

B.2 Downstream Experimental Settings

During downstream finetuning, we load the pretrained weights of UniBCI and replace the original reconstruction head with a task-specific head composed of a multi-layer perceptron (MLP). The learned representations are flattened and fed into this head for downstream classification or regression tasks. We then perform full-parameter finetuning on the target datasets while freezing the text encoder. For classification tasks, we use the cross-entropy loss, whereas for regression tasks, we adopt the mean squared error (MSE) loss. All downstream results are reported as the mean $\pm$ standard deviation over five random seeds, ranging from 42 to 46.

C Theoretical Proof

C.1 Metadata Injection Expands the Semantic Space of Spike Tokens

Let the spike embedding after projection be

$$X_s \in \mathbb{R}^{n \times d}, \tag{13}$$

where d is the hidden dimension. Due to the sparsity and binary event-driven nature of invasive spike trains, the intrinsic dimensionality of X_s is much smaller than the ambient dimension:

$$\operatorname{rank}(X_s) = r_s \ll d. \tag{14}$$

Equivalently, its covariance matrix

$$\Sigma_s = \frac{1}{n} X_s^\top X_s \tag{15}$$

is low-rank, and its differential entropy satisfies

$$H(X_s) \leq \frac{1}{2} \log \det(2\pi e \Sigma_s). \tag{16}$$

Since most eigenvalues of Σ_s are close to zero, we have

$$\det(\Sigma_s) \to 0, \tag{17}$$

which implies latent crowding and weak inter-source separability.

Let metadata be encoded by a pretrained language model as

$$X_m \in \mathbb{R}^{n\times d}, \tag{18}$$

whose semantic representation spans a high-rank manifold:

$$\mathrm{rank}(X_m) = r_m, \quad r_m \approx d. \tag{19}$$

The final token representation is

$$Z = X_s + X_m. \tag{20}$$

Assuming weak cross-correlation between spike features and metadata semantics,

$$\Sigma_Z = \Sigma_s + \Sigma_m. \tag{21}$$

By Minkowski determinant inequality,

$$\det(\Sigma_Z)^{1/d} \geq \det(\Sigma_s)^{1/d} + \det(\Sigma_m)^{1/d}, \tag{22}$$

thus

$$\det(\Sigma_Z) > \det(\Sigma_s), \quad H(Z) > H(X_s). \tag{23}$$

For source discrimination, consider two samples i, j:

$$\|Z_i - Z_j\|_2^2 = \|X_i - X_j\|_2^2 + \|M_i - M_j\|_2^2 + 2\langle X_i - X_j, M_i - M_j\rangle. \tag{24}$$

Since semantic metadata embeddings are approximately orthogonal,

$$\mathbb{E}\big[\langle X_i - X_j, M_i - M_j\rangle\big] \approx 0, \tag{25}$$

we obtain

$$\mathbb{E}\|Z_i - Z_j\|_2^2 > \mathbb{E}\|X_i - X_j\|_2^2. \tag{26}$$

C.2 Advantage of Linear Attention for Spike Signals

Given input $H \in \mathbb{R}^{t\times d}$, standard attention computes pairwise temporal affinities:

$$\mathrm{Attn}(H) = \mathrm{softmax}(QK^\top)V, \quad Q = HW_q,\ K = HW_k,\ V = HW_v. \tag{27}$$

Linear attention replaces softmax with an explicit feature map $\phi(\cdot) = \mathrm{elu}(\cdot) + 1$:

$$\mathrm{Attn}(H) = \phi(Q)\left(\phi(K)^\top V\right). \tag{28}$$

For spike signals $H = S + \epsilon$, with sparse component S and zero-mean noise ϵ, standard attention computes $t \times t$ pairwise interactions:

$$QK^\top \approx SS^\top + S\epsilon^\top + \epsilon S^\top + \epsilon\epsilon^\top. \tag{29}$$

At inactive steps ($S \approx 0$), instantaneous noise cross-terms $\epsilon\epsilon^\top$ dominate and are exponentially amplified by softmax.

In contrast, linear attention aggregates temporal context prior to query interaction. Applying a first-order expansion $\phi(K) \approx \phi(S) + \phi'(S) \odot \epsilon$ yields:

$$\phi(K)^\top V = \sum_{\tau=1}^{t} \phi(K_\tau)^\top V_\tau \approx \phi(S)^\top S + \sum_{\tau=1}^{t} \Big(\phi(S_\tau)^\top \epsilon_\tau + (\phi'(S_\tau) \odot \epsilon_\tau)^\top S_\tau\Big). \tag{30}$$

Since ϵ is zero-mean and temporally uncorrelated, noise terms accumulate incoherently as $\mathcal{O}(\sqrt{t})$, whereas sparse signal terms $\phi(S)^\top S$ accumulate coherently over active steps. The non-negativity of ϕ prevents sign cancellation, allowing queries to interact with a denoised temporal summary and providing a more stable co-activation estimator. This makes linear attention better suited for handling sparse spike data.

C.3 Shared Manifold Learning via Interval-Area Attention

Let $H_{i,a} \in \mathbb{R}^{t \times d}$ denote the representation of interval i and area a. We assume neural activity is generated from a shared latent dynamical state $z_i \in \mathbb{R}^{d_z}$:

$$H_{i,a} = f_a(z_i) + \epsilon_{i,a}, \tag{31}$$

where $f_a : \mathbb{R}^{d_z} \to \mathbb{R}^{t \times d}$ is an area-specific smooth mapping and $\epsilon_{i,a}$ is zero-mean noise with bounded variance.

Interval-wise Encoding. The interval attention module applies a transformation $g(\cdot)$ to extract temporal features:

$$\tilde{O}_{i,a} = g(H_{i,a}). \tag{32}$$

Under the assumption that $g(\cdot)$ is Lipschitz continuous and approximately preserves local structure, we have:

$$\tilde{O}_{i,a} \approx \tilde{f}_a(z_i) + \tilde{\epsilon}_{i,a}, \tag{33}$$

where $\tilde{f}_a$ is a transformed mapping and $\tilde{\epsilon}_{i,a}$ remains bounded.

Area-wise Alignment via Attention. After temporal pooling, denote $\bar{H}_{i,a} \in \mathbb{R}^d$ as the aggregated representation. The area-wise attention computes:

$$O_{i,a} = \sum_{(j,b) \in \mathcal{N}(i,a)} \alpha_{(i,a),(j,b)}\, \bar{H}_{j,b}, \tag{34}$$

where $\mathcal{N}(i,a)$ is the sliding window neighborhood and α are attention weights.

Substituting the generative model:

$$O_{i,a} \approx \sum_{(j,b)} \alpha_{(i,a),(j,b)}\, f_b(z_j). \tag{35}$$

If the latent dynamics are smooth across nearby intervals, i.e.,

$$\|z_i - z_j\| \le \delta \quad \text{for } (j,b) \in \mathcal{N}(i,a), \tag{36}$$

then by smoothness of f_b:

$$f_b(z_j) \approx f_b(z_i). \tag{37}$$

Thus, the aggregation simplifies to:

$$O_{i,a} \approx \sum_b \beta_{a,b} f_b(z_i), \tag{38}$$

which is a weighted combination of mappings evaluated at the same latent state z_i.

Emergence of a Shared Manifold. Define a unified representation:

$$\phi(z_i) = \sum_b \beta_{a,b} f_b(z_i). \tag{39}$$

Then:

$$O_{i,a} \approx \phi(z_i), \tag{40}$$

which is independent of the specific area a up to weighting coefficients.

Therefore, all representations $\{O_{i,a}\}_a$ lie near a shared manifold:

$$\mathcal{M} = \{\phi(z) \mid z \in \mathbb{R}^{d_z}\}. \tag{41}$$

Stacked Refinement. With multiple layers, the residual updates:

$$H^{\ell+1} = \mathcal{F}(H^\ell) + H^\ell \tag{42}$$

iteratively reduce the variance of $\epsilon_{i,a}$ and enforce consistency across areas. This can be interpreted as a contraction mapping toward $\mathcal{M}$, leading to progressively aligned representations.

Conclusion. Interval-Area Attention acts as a structured operator that (i) preserves local temporal dynamics, and (ii) enforces cross-area consistency through attention-based aggregation. Under mild smoothness assumptions, this results in learning a shared latent manifold that explains neural activity across both spatial areas and temporal intervals.

D COMPUTATIONAL COMPLEXITY ANALYSIS

We analyze the computational complexity of the proposed Interval-Area Attention (IAA), which consists of interval linear attention (ILA) and area-wise sliding window attention (ASWA). Let the input feature be $H \in \mathbb{R}^{N\times A\times t\times d}$, where N denotes the number of temporal intervals, A the number of grouped areas, t the token length within each interval, and d the embedding dimension.

D.1 INTERVAL LINEAR ATTENTION

For each interval-area pair $H_{i,a} \in \mathbb{R}^{t\times d}$, ILA applies linear attention independently:

$$\tilde{O}_{i,a} = H_{i,a}W_q \cdot \left((H_{i,a}W_k)^\top H_{i,a}W_v\right). \tag{43}$$

Unlike standard self-attention with quadratic complexity $\mathcal{O}(t^2d)$, linear attention first computes the key-value interaction $(K^\top V)$ with complexity $\mathcal{O}(td^2)$, followed by query projection with complexity $\mathcal{O}(td^2)$. Therefore, the complexity for one interval-area pair is $\mathcal{O}(td^2)$. Since computation is performed over all $N \times A$ interval-area pairs, the total complexity of ILA is $\mathcal{O}(NAtd^2)$.

D.2 AREA-WISE SLIDING WINDOW ATTENTION

After temporal pooling, the token sequence is reduced to $S = N \times A$, and the feature of spike tokens denotes as $\tilde{H} \in \mathbb{R}^{S\times d}$. Standard global self-attention across all interval-area tokens requires:

$$\mathcal{O}(S^2d). \tag{44}$$

To avoid quadratic growth, ASWA restricts each query to a local sliding window of size w. Thus, each token only attends to its neighboring w tokens, and the total complexity is reduced to $\mathcal{O}(Swd)$. Since w is constant and independent of sequence length, the complexity becomes approximately linear with respect to S: $\mathcal{O}(Sd)$.

E CODE ACCESS

The code of this paper is accessible at: `https://anonymous.4open.science/r/UniBCI-C805`.

F DATASETS

F.1 PRETRAINING DATASETS

F.1.1 M1-CO1 DATASET

The M1-CO1 dataset records neural activity from a macaque performing a standard eight-direction center-out reaching task. In this task, the subject moves a cursor from the center toward one of eight peripheral targets based on visual cues. Unlike continuous control tasks, the labels are discrete movement directions, represented as categorical classes ranging from 0 to 7. Neural signals were recorded from the primary motor cortex (M1).

Neural and Behavioral Signals: The dataset is stored in HDF5 format and contains sorted spike trains. Each file corresponds to a single trial. The recordings include up to 232 neural units, providing high-resolution population activity. The behavioral label for each trial is a single discrete direction index.

Preprocessing: To standardize inputs for the model, variable-length spike trains are resampled using a temporal binning strategy. Specifically, an adaptive binning approach is applied to divide each

trial into 100 equal segments along the temporal axis. Spike events within each bin are aggregated to form spike count sequences, resulting in a fixed-length representation of 100 time steps. During pretraining, a total of 348 sessions are used.

F.1.2 M1-CO2 DATASET

The M1-CO2 dataset records neural and behavioral activity from a macaque performing an eight-direction center-out reaching task in a whack-a-mole paradigm using a robotic interface. Neural signals were recorded from the contralateral primary motor cortex (M1) via a 64-channel electrode array. The dataset consists of 29 recording days collected in April 2025, comprising a total of 32,083 valid trials.

Neural and Behavioral Signals: Raw neural signals were recorded at a sampling rate of 30 kHz across 64 channels. Behavioral signals, consisting of two-dimensional joystick voltage trajectories, were simultaneously recorded at 1 kHz.

Preprocessing: Spike detection was first applied to convert continuous neural voltage signals into discrete spike events, forming a two-dimensional spike matrix. To ensure temporal alignment, behavioral signals were linearly interpolated to 30 kHz and synchronized with neural recordings. Both neural and behavioral data were then downsampled to 50 Hz using temporal binning. Gaussian smoothing was applied to the behavioral trajectories to improve the signal-to-noise ratio. Finally, the data were segmented into fixed-length samples of 2 seconds, corresponding to 100 time steps, forming 64-dimensional spike sequence inputs paired with 2D trajectory decoding targets.

F.1.3 PAC-MAN DATASET

The Pac-Man dataset records neural and behavioral activity from a macaque performing a Pacman-based continuous tracking task. The subject controls an on-screen character using a joystick to complete the task. Neural signals were recorded from multiple brain regions involved in motor planning and execution, primarily the premotor cortex (PMd), as well as the dorsolateral prefrontal cortex (dlPFC) and anterior cingulate cortex (ACC). The dataset consists of continuous recording sessions spanning multiple days. The sampling rate is 60 Hz.

Neural and Behavioral Signals: The dataset contains sorted single-unit spike trains and synchronized behavioral trajectories. Due to variations in recording conditions across days, the number and anatomical locations of identified neurons vary dynamically. Behavioral data consist of two-dimensional position coordinates (X, Y) of the Pacman agent during task execution.

Preprocessing: Neural spike trains and behavioral trajectories are first temporally aligned across brain regions. To smooth neural firing rates and extract temporal features, a 100 ms binning strategy is applied, and behavioral positions are averaged within each bin. A sliding window approach is then used to segment continuous recordings into samples, with each window spanning 10 seconds (100 time bins). Data augmentation is performed using a 2-second stride (20 time bins). During pretraining, recordings spanning 121 days are used.

F.1.4 HPC-HG DATASET

The HPC-HG dataset records neural activity from a macaque (Monkey P) performing a free-moving foraging task in a large 3D arena (3.5 m $\times$ 3.5 m $\times$ 2.8 m). The subject is required to search for hidden reward locations. Once the animal enters a reward zone, it must visit a randomly illuminated touchscreen located around the arena to obtain food rewards. The reward location is shifted after 50 successful retrievals.

Neural and Behavioral Signals: Neural signals were recorded from the hippocampus (HPC), which is closely associated with spatial representation and memory. The dataset contains spike trains from multiple sorted single units, sampled at 100 Hz (10 ms resolution). Due to cross-day variability, the number of identified units varies across sessions. Behavioral data consist of synchronized two-dimensional spatial coordinates (X, Y) of the animal within the arena.

Preprocessing: Invalid behavioral samples (NaN values) and their corresponding neural segments are removed using a synchronization mask to ensure precise alignment between hippocampal activity and spatial trajectories. Spike trains are then binned using a 100 ms window, and spike counts

are accumulated within each bin. The corresponding spatial coordinates are averaged over the same temporal bins. A sliding window approach is applied to segment the continuous recordings into samples, each containing 10 seconds of data (100 time bins), with a stride of 2 seconds for data augmentation. During pretraining, recordings spanning 36 days are used.

F.1.5 LICK DATASET

The LICK dataset records neural activity and licking behavior in water-deprived mice performing a periodic water-reward task. The dataset includes 14 adult male mice with a recording duration of 119 days and a total of 28,573 trials. The animals are head-fixed and trained to perform a fixed-interval licking task in which a sucrose reward of 2 to 3 μL is delivered every 10 seconds. Neural signals were recorded from three brain regions including secondary motor cortex (M2) in 5 mice, substantia nigra pars reticulata (SNR), and ventrolateral striatum (VLS) in 9 mice.

Neural and Behavioral Signals: Neural activity was recorded using a 32-channel Cerebus system at 30 kHz sampling rate. Behavioral signals were simultaneously recorded using electrical or infrared sensors and converted into a binary sequence, where 0 indicates no licking and 1 indicates licking. The released dataset is downsampled to 2,000 Hz to ensure precise temporal alignment between neural spikes and behavioral events.

Preprocessing: The preprocessing pipeline applies both binning and sliding window segmentation. First, the data are downsampled from 2,000 Hz to 100 Hz by accumulating spike counts within each bin. Behavioral labels are aggregated using a max operation, such that any licking event within a bin is assigned a label of 1, ensuring precise alignment between neural dynamics and discrete behavioral events. A sliding window strategy is then used to segment the continuous recordings, with a fixed window length of 100 time steps and a stride of 95. During pretraining, recordings from 11 mice across 104 days are used, forming fixed-dimensional neural spike sequences paired with binary behavioral prediction targets.

F.2 DOWNSTREAM DATASETS

F.2.1 M1-CO1 DATASET

For downstream evaluation, we use 24 sessions spanning 10 days that were not included in pre-training. The task is an eight-class classification of movement directions. We evaluate M1-CO1 under two protocols: (1) *Multi-day*, where all sessions across the 10 recording days are pooled and randomly split at the session level with an 80%/20% ratio for training and testing, ensuring that both sets contain trials from all directions; and (2) *Cross-day*, where data from 8 days are used for training and the remaining 2 days are held out for testing to assess temporal generalization across recording days. The neural and behavioral signals, as well as preprocessing steps, follow the same procedures described in the pretraining section.

F.2.2 LICK DATASET

For downstream evaluation, we use data from three mice that are not seen during pretraining, with one mouse selected from each brain region. For each mouse, the data are randomly split into training and testing sets with an 80% to 20% ratio. The task is formulated as a binary classification problem of licking behavior prediction. The final performance is reported by averaging the results across the three mice.

F.2.3 VISUAL-CODING DATASET

The Visual-Coding dataset[1] contains extracellular neural recordings from mice performing an 8-class visual decoding task during drifting gratings presentation (Siegle et al., 2021). Neural activity was recorded simultaneously from multiple brain structures using high-density Neuropixels probes. For our benchmark, the first 17 sessions from 17 mice were selected, with units in the primary visual cortex (VISp) targeted for visual feature decoding. In each trial, one of eight drifting grating directions was presented on a screen, capturing neural activity during visual feature encoding and direction processing.

[1] https://dandiarchive.org/dandiset/000021/0.251116.2246

Neural and Behavioral Signals: The dataset provides sorted spike trains from recorded units, together with trial-level annotations including 8-class orientation labels, temporal boundaries (start/end of each trial), and spatial probe mapping (electrodes and brain regions).

Preprocessing: Spike times are extracted from NWB files and aligned to trial events. Neurons are filtered based on brain area (targeting VISp) and quality metrics. Signals are binned into fixed-length temporal windows to form spike count tensors of shape $[\mathrm{Trials}, \mathrm{Time}, \mathrm{Channels}]$. The data are then split into training and testing sets with an 80%/20% ratio. The task is formulated as an 8-class classification problem for decoding visual motion direction from cortical neural activity.

F.2.4 PPC-FINGER DATASET

The PPC-FINGER dataset[2] contains invasive neural recordings from a single tetraplegic participant performing a 6-class BCI finger press task. Neural activity was recorded from the left posterior parietal cortex (PPC, PC-IP region) using a 96-channel Utah array (Blackrock Microsystems). In each trial, one of six finger cues was presented at a random screen location, and the participant attempted to press the corresponding finger following visual instruction. Each trial lasts approximately 1.5 s after cue onset, capturing neural activity during motor intention and execution planning.

Neural and Behavioral Signals: The dataset provides sorted spike trains from all channels, together with trial-level annotations including 6-class finger labels, cue locations, and precise temporal boundaries (start/end of each trial).

Preprocessing: Spike times are extracted from NWB files and aligned to trial events using the recorded sampling rate. Signals are binned into fixed-length temporal windows to form spike count tensors of shape $[\mathrm{Trials}, \mathrm{Time}, \mathrm{Channels}]$. The data are then split into training and testing sets with an 80%/20% ratio. The task is formulated as a 6-class classification problem for decoding finger movement intentions from PPC activity.

F.2.5 PERICH DATASET

The Perich dataset[3] consists of 117 recording sessions collected from four non-human primates, denoted as Monkey C, M, J, and T, covering a range of motor control and planning tasks. The subjects are seated and control a cursor on a two-dimensional screen using a joystick. The tasks include two paradigms. The center-out task involves reaching from a central position to one of eight targets arranged on a circle with a radius of 8 cm. Each trial includes a hold period, a delay until a go cue, and a subsequent reaching movement. The random target task involves continuously moving between randomly placed targets instead of fixed circular positions.

Neural and Behavioral Signals: Neural activity was recorded using chronic multi-electrode arrays implanted in the primary motor cortex (M1) and dorsal premotor cortex (PMd). The raw data were acquired using the Blackrock Cerebus system and processed with single-unit spike sorting. Channels with high crosstalk and units with excessive coincident spikes were removed to ensure neuron independence.

Preprocessing: For downstream evaluation, we use 12 sessions from Monkey T, including 6 center-out sessions and 6 random target sessions, which are completely held out from pretraining. Data preprocessing, binning, and dataset splitting are performed using the brainsets toolkit provided in (Azabou et al., 2023), ensuring a standardized and reproducible pipeline. The data are randomly split into training and testing sets with an 80% to 20% ratio. Neural signals are binned with a temporal resolution of 10 ms to obtain spike count sequences. The data are then segmented into fixed-length sequences of 1.0 second, corresponding to 100 time steps, forming standardized samples for motor decoding tasks.

F.2.6 MC-MAZE DATASET

The MC-Maze dataset[4] is derived from the Neural Latent Benchmark and records motor cortical activity from a macaque performing an obstacle-constrained maze reaching task. The subject performs

[2] https://doi.org/10.48324/dandi.000147/0.221122.2256
[3] https://doi.org/10.48324/dandi.000688/0.250122.1735
[4] https://doi.org/10.48324/dandi.000128/0.220113.0400

a delayed center-out reaching task with obstacle constraints. In this task, the macaque must navigate around obstacles on the screen to reach a target. Due to varying obstacle configurations, the task generates highly stereotyped linear and curved movement trajectories. The experiment simultaneously records cursor position, eye gaze, and two-dimensional hand position and velocity.

Neural and Behavioral Signals: Neural activity was recorded from the dorsal premotor cortex (PMd) and primary motor cortex (M1). Prior studies have shown that neural dynamics in M1 and PMd exhibit strong predictability around movement onset, reflecting pronounced autonomous dynamics.

Preprocessing: The dataset is standardized for dynamical modeling. Neural signals are processed using a temporal resolution of 7 ms. We follow the data acquisition and preprocessing pipeline provided in (Song et al., 2025) for dataset downloading and processing. The dataset uses the default train-test partitioning released with the original data, corresponding to an approximate 75%/25% split. The behavioral targets are the two-dimensional hand velocities (v_x, v_y), and the task is formulated as a regression problem to evaluate the model's ability to extract stable neural representations under limited data conditions.

F.2.7 AREA2-BUMP DATASET

The Area2-Bump dataset[5] is derived from the Neural Latent Benchmark and records somatosensory cortical activity from a macaque performing a center-out reaching task with mechanical perturbations. The task follows a standard center-out reaching paradigm, with additional mechanical perturbations applied in a subset of trials. Specifically, the subject's arm is subjected to random force perturbations before movement onset, requiring the animal to adjust its trajectory to reach the target. The experiment records rich multimodal kinematic signals, including hand position, velocity, acceleration, interaction forces between the hand and joystick, as well as muscle lengths, muscle velocities, joint angles, and joint velocities.

Neural and Behavioral Signals: Neural activity was recorded from the somatosensory cortex (S1), specifically Brodmann area 2. Unlike motor cortical regions that exhibit strong autonomous dynamics, neural responses in this area are strongly driven by external perturbations and encode detailed whole-arm kinematics.

Preprocessing: The dataset is standardized for dynamical modeling. Neural signals are processed with a temporal resolution of 6 ms. We follow the data acquisition and preprocessing pipeline provided in (Song et al., 2025) for dataset downloading and processing. The dataset uses the default train-test partitioning released with the original data, corresponding to an approximate 75%/25% split. Behavioral targets are the hand velocity signals, and the task is formulated as a regression problem to evaluate the model's ability to learn neural representations under complex non-autonomous sensory inputs.

G BASELINES IMPLEMENTATION

G.1 FROM-SCRATCH BASELINES.

WF (Glaser et al., 2020): We adopt the Wiener filter (WF) as a traditional linear decoding baseline. WF models the mapping from neural spike activity to behavioral outputs using a linear regression framework with temporal filtering. It serves as a strong classical benchmark for neural decoding tasks and is trained directly on each downstream task without any pretraining. For fair comparison, the same data splits and evaluation protocols are used as in all other methods.

GRU (Cho et al., 2014): As a sequential modeling baseline, we implement a gated recurrent unit (GRU) decoder to capture temporal dependencies in neural spike sequences. The input spike tensor is fed into a multi-layer GRU, followed by a linear prediction head for classification or regression tasks. The GRU is trained end-to-end directly on downstream tasks without self-supervised pretraining. Its training setup, including data splitting, optimizer, and learning rate, is kept consistent with other baselines for fair comparison.

[5] https://doi.org/10.48324/dandi.000127/0.220113.0359

MLP: As a simple baseline, we implement a multi-layer perceptron (MLP) that directly maps the input neural spike tensor to the target output. Unlike other pretrained models, the MLP is not pretrained in a self-supervised manner and is trained end-to-end directly on the downstream task. For a fair comparison, its training setup, including data splitting and optimization strategy, is kept consistent with that of other methods.

NoMAD (Karpowicz et al., 2025): NoMAD trains the dynamical model and decoder on a reference session, and then learns an alignment network in an unsupervised manner on subsequent sessions to map new neural data onto the fixed dynamical model of the reference session. In this work, tailored to downstream tasks, we use 60% of the data to train the dynamical model and decoder in a supervised manner, followed by 20% of the data for further unsupervised alignment, and reserve the remaining 20% for evaluation.

G.2 Re-pretrained Baselines.

VAE (Kingma & Welling, 2013): To provide a generative modeling baseline, we implement a Variational Autoencoder (VAE) tailored for neural spike data. Given an input spike tensor, we first flatten it into a vector and feed it into a multi-layer perceptron (MLP)-based encoder. The encoder consists of two fully connected layers with Layer Normalization and GELU activations, which output the mean μ and log-variance $\log \sigma^2$ of the latent distribution. A latent variable z is then sampled via the reparameterization trick, and the decoder, implemented as another MLP, reconstructs the input from the latent representation. The model is optimized with the objective

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + \beta \cdot \mathcal{L}_{\text{KL}} \tag{45}$$

where $\mathcal{L}_{\text{recon}}$ denotes the reconstruction loss and $\mathcal{L}_{\text{KL}}$ denotes the Kullback–Leibler divergence. For a fair comparison, the VAE baseline is trained under the same pretraining protocol as our proposed model, including the same large-scale neural spike dataset, identical training schedule (optimizer, learning rate, and number of epochs), and a fully self-supervised learning paradigm without any label information. We use the AdamW optimizer with cosine annealing learning rate scheduling. The reconstruction objective is implemented using mean squared error (MSE) loss, which is empirically more stable for spike signal reconstruction. A small weighting factor $\beta = 10^{-5}$ is applied to the KL divergence term to balance representation learning and reconstruction fidelity.

NDT1 (Ye & Pandarinath, 2021): Neural Data Transformer (NDT) is a non-recurrent framework for modeling neural population activity based on a Transformer encoder architecture. Unlike traditional recurrent neural networks (RNNs), NDT employs self-attention to process neural spike sequences in parallel and captures latent neural dynamics through a masked modeling objective. In this work, we re-pretrain NDT1 using the NDT1 implementation provided in the MtM (Zhang et al., 2024) codebase, on the same pretraining corpus as UniBCI and following the original NDT1 configuration. The pretrained NDT1 encoder is equipped with a task-specific MLP head for downstream prediction.

NDT2 (Ye et al., 2023): Neural Data Transformer 2 (NDT2) is an extension of the original NDT architecture, designed to address distribution shifts in neural signals across different experiments and subjects through multi-context pretraining. Compared to its predecessor, NDT2 adopts a more powerful Transformer architecture and supports joint training across diverse tasks and recording conditions, enabling the model to learn more generalizable representations of neural population dynamics. In this work, we re-pretrain NDT2 using the NDT2 implementation provided in the MtM (Zhang et al., 2024) codebase, on the same pretraining corpus as UniBCI and following the original NDT2 configuration. The pretrained NDT2 encoder is then paired with a task-specific MLP head for downstream prediction.

MtM (Zhang et al., 2024): Multi-task Masking (MtM) is a self-supervised framework for modeling neural population activity, which learns multi-scale representations by alternately performing masking and reconstruction tasks across different spatial scales (single-neuron, intra-region, and inter-region) and temporal dimensions. By introducing learnable prompt tokens for task switching, MtM enables a unified architecture to jointly optimize multiple objectives, including neural activity reconstruction, forward prediction, and inter-region interaction modeling. In our experiments, we re-pretrain the MtM model built upon the NDT2 (Ye et al., 2023) architecture on the same corpus as UniBCI, following the original training configuration. Since our pretraining data do not provide

fine-grained region annotations, we remove region masking while retaining causal masking and neuron masking. The pretrained model is subsequently used as the encoder with a task-specific MLP head for downstream prediction.

POYO (Azabou et al., 2023): POYO (Pre-training On manY neurOns) is a unified framework for neural population decoding, which represents individual spike events as discrete tokens consisting of neuron embeddings and timestamps. Built upon the PerceiverIO architecture with rotary positional encoding, POYO enables scalable modeling of neural recordings across animals and sessions. In contrast to the original POYO setting, which is pretrained for a single behavioral task, our pretraining corpus contains multiple tasks. We therefore adapt POYO to a multi-task setting by introducing task-specific prediction heads, following the multi-task formulation used in POYO+ Azabou et al. (2025). The resulting model is re-pretrained on the same corpus as UniBCI and subsequently fine-tuned on downstream tasks.

POYO+ (Azabou et al., 2025): POYO+ is a multi-task and multi-session Transformer-based neural decoder designed to address the challenges of high heterogeneity across large-scale neural recordings. Extending the tokenization framework of POYO, POYO+ introduces task-specific query tokens with session and task embeddings to unify continuous regression, sequence classification, and segmentation tasks within a shared cross-attention latent space. In our experiments, we re-pretrain POYO+ on the same corpus as UniBCI using its original multi-task formulation. Thus, POYO and POYO+ share the same multi-task training setup in our experiments, while POYO+ additionally conditions its query tokens on task embeddings.

Others: Other baseline models, including POSSM (Ryoo et al., 2025), NEDS (Zhang et al., 2025), and NDT3 (Ye et al., 2025), are not included in our comparison due to the lack of publicly available code or the inability to establish a standardized re-pretraining setup under our experimental protocol. We note that this constraint is common in the literature, and we instead compare against a set of widely adopted and reproducible baselines.

G.3 Official Pretrained Checkpoints.

In addition to the controlled re-pretraining setting described above, we further evaluate the officially released pretrained checkpoints of NDT1, NDT2, MtM, and POYO in Appendix I.1. In this setting, we directly use the publicly released weights without additional pretraining on our corpus, providing a complementary evaluation of their off-the-shelf performance.

H METRICS DESCRIPTION

We evaluate the proposed method on both classification and regression tasks using a comprehensive set of metrics. For classification, we adopt: 1) **Balanced Accuracy**, defined as the average recall across all classes, making it suitable for both binary and multi-class settings, especially under class imbalance, and 2) **Weighted F1 score**, the harmonic mean of precision and recall, weighted by class support to provide a balanced evaluation across classes. For regression, we report the coefficient of determination (R^2), which measures the proportion of variance in the target variable explained by the model.

I ADDITIONAL RESULTS

I.1 Evaluation with Official Pretrained Checkpoints

Table 4 reports the downstream performance of NDT1, NDT2, MtM, and POYO using their officially released pretrained checkpoints, with all experiments conducted under a fixed random seed of 42. In contrast to Table 2, where all methods are pretrained on the same standardized corpus under a unified protocol for controlled architectural comparison, Table 4 evaluates the official pretrained models directly without additional pretraining. UniBCI achieves the best overall performance in both settings, outperforming the official checkpoints on most classification and regression tasks, with an average classification accuracy of 0.854 and an average regression R^2 of 0.811. Taken together, Tables 2 and 4 demonstrate that UniBCI consistently outperforms existing methods under both controlled and off-the-shelf evaluation settings.

Table 4: Performance comparison using officially released pretrained checkpoints on classification and regression tasks.

Generalization	Dataset	Metrics	NDT1	NDT2	MtM	POYO	UniBCI (Ours)
Classification Tasks							
New Session	M1-CO1 (Multi-day)	B-Acc	0.755	0.841	0.842	0.875	**0.909**
		F1-W	0.756	0.841	0.844	0.876	**0.910**
	M1-CO1 (Cross-day)	B-Acc	0.816	0.829	0.825	0.842	**0.853**
		F1-W	0.819	0.832	0.828	0.839	**0.853**
New Subject	LICK	B-Acc	0.721	0.730	0.734	0.730	**0.757**
		F1-W	0.858	**0.862**	0.861	0.857	0.859
New Specie	PPC-FINGER	B-Acc	0.959	0.961	0.934	0.940	**0.968**
		F1-W	0.958	0.959	0.932	0.939	**0.968**
New Task	Visual-Coding	B-Acc	**0.784**	0.781	0.754	0.718	0.782
		F1-W	0.782	0.781	0.749	0.719	**0.783**
		Avg. B-Acc	0.807	0.828	0.818	0.821	**0.854**
Regression Tasks							
New Subject	Perich T-CO	R^2	0.737	0.738	0.682	0.753	**0.765**
New Task	Perich T-RT	R^2	0.687	0.681	0.621	0.692	**0.698**
	MC-Maze	R^2	0.882	0.884	0.879	0.883	**0.890**
New Region	Area2-Bump	R^2	**0.902**	0.888	0.874	0.882	0.890
		Avg. R^2	0.802	0.798	0.764	0.803	**0.811**

I.2 Few-shot Evaluation

To further evaluate the data efficiency of the proposed model, we conduct few-shot experiments by limiting the amount of labeled data available during training. Specifically, we use only 20% of the data for training and reserve the remaining 80% for testing. This setting simulates realistic scenarios in neural data analysis, where obtaining large-scale labeled data is expensive and time-consuming.

The results on M1-CO1 and LICK datasets are summarized in Table 5. Compared to the standard setting, all methods experience performance degradation due to the limited supervision. However, UniBCI consistently outperforms all baselines by a clear margin. On the M1-CO1 dataset, UniBCI achieves a balanced accuracy of 0.782 and a weighted F1 score of 0.781, surpassing the previous best method POYO. On the LICK dataset, UniBCI achieves the highest balanced accuracy of 0.732 and the best weighted F1 score of 0.839. These results demonstrate that UniBCI maintains strong discriminative ability even with severely limited labeled data.

We further evaluate the model on more challenging cross-dataset transfer tasks, as shown in Table 6. Under the few-shot setting, the performance gap between methods becomes more pronounced. UniBCI achieves the best performance on Perich T-CO and competitive results on Perich T-RT, while maintaining robust performance on MC-Maze and Area2-Bump. Overall, UniBCI achieves the highest average score of 0.715, outperforming all competing methods. Notably, while some baselines exhibit performance drops under limited data, UniBCI shows more stable degradation, indicating stronger robustness.

I.3 Ablation Study

We conduct an ablation study on two key design choices of the proposed model: positional encoding and token grouping strategy. Specifically, we compare STPE (spatio-temporal absolute positional encoding) with RoPE (rotary positional encoding), and evaluate two grouping strategies: Uniform Group and Direct Group. For grouping, Uniform Group first evenly partitions channels (or units) into a predefined number of areas (*Area Number*), and pads each group to a fixed size (*Area Size*) when necessary, ensuring balanced and consistent token structures. In contrast, Direct Group directly pads all channels to the maximum length and then splits them into groups according to *Area*

Table 5: Performance comparison on M1-CO1 and LICK datasets.

Method	M1-CO1 (Multi-day)		LICK	
	Balanced Accuracy	Weighted F1	Balanced Accuracy	Weighted F1
WF	0.372	0.415	0.550	0.505
GRU	0.410	0.435	0.519	0.509
MLP	0.499	0.496	0.609	0.811
VAE	0.576	0.572	0.638	0.820
NDT1	0.509	0.503	0.660	0.825
NDT2	0.725	0.721	0.686	0.834
MtM	0.727	0.725	0.671	0.829
POYO	<u>0.762</u>	<u>0.763</u>	<u>0.687</u>	<u>0.835</u>
UniBCI	**0.782**	**0.781**	**0.732**	**0.839**

Table 6: Performance comparison of different methods.

Method	Perich T-CO	Perich T-RT	MC-Maze	Area2-Bump	Avg. R^2
WF	0.372	0.415	0.550	0.505	0.461
GRU	0.410	0.435	0.519	0.509	0.468
MLP	0.610	0.509	0.704	0.507	0.583
VAE	0.634	0.511	0.813	0.704	0.666
NDT1	0.636	0.528	0.862	**0.814**	0.710
NDT2	0.643	0.543	<u>0.867</u>	0.793	<u>0.712</u>
MtM	0.584	0.512	**0.871**	0.761	0.682
POYO	<u>0.650</u>	**0.568**	0.846	0.742	0.702
UniBCI	**0.653**	<u>0.562</u>	0.846	<u>0.798</u>	**0.715**

Number, which may result in some groups containing mostly or entirely zero-padding tokens. As shown in Table 7, STPE consistently provides more stable performance across tasks than RoPE. Although RoPE achieves comparable classification results on M1-CO1, it leads to clear performance degradation on the regression tasks MC-Maze and Area2-Bump, indicating that explicit temporal indexing is more suitable for neural spike representations. For grouping strategy, Uniform Group generally outperforms Direct Group by maintaining more regular token distributions and reducing redundant padded regions. Direct Group introduces irregular structures and may generate empty groups, which weakens representation consistency and harms overall robustness, despite showing competitive results on Area2-Bump. Overall, the combination of STPE and Uniform Group achieves the best trade-off between accuracy, robustness, and efficiency, validating its effectiveness for neural signal modeling.

Table 7: Ablation study on positional encoding and grouping strategy.

Variants	M1-CO1 (Multi-day)		MC-Maze (R^2)	Area2-Bump (R^2)
	Balanced Accuracy	Weighted F1		
STPE + Uniform Group	**0.909**	**0.910**	**0.890**	0.890
RoPE + Uniform Group	0.893	0.894	0.850	0.840
STPE + Direct Group	0.834	0.833	0.879	**0.892**

I.4 SENSITIVITY ANALYSIS

To further evaluate the robustness of the proposed model, we conduct a comprehensive sensitivity analysis on key tokenization hyperparameters, including Interval Size (t), Area Size (C_a), Area Number (A), and sliding window size (w), with results summarized in Table 8. We observe that the model is highly sensitive to temporal resolution: an excessively small interval ($t = 2$) leads to a significant performance drop (e.g., Balanced Accuracy 0.670 on M1-CO1), likely due to fragmentation of continuous neural dynamics that impairs global dependency modeling within limited attention contexts, while an overly large interval ($t = 20$) also degrades performance by discarding critical high-frequency temporal information; in contrast, the default setting ($t = 10$) achieves the best trade-off between temporal fidelity and efficiency. For Area Size (C_a), increasing C_a from 8 to 32 consistently improves performance, suggesting enhanced modeling of population-level features within each token, whereas further increasing to $C_a = 64$ yields marginal gains or slight degradation, indicating that overly large areas may introduce redundancy and obscure distinct neural contributions, highlighting the importance of mesoscopic-scale representations. Furthermore, the interaction between Area Number (A) and sliding window size (w) exhibits dataset-dependent behavior: for example, a compact configuration ($A = 2, w = 4$) achieves the best performance on Area2-Bump ($R^2 = 0.901$), while more complex tasks such as MC-Maze benefit from larger A, suggesting that different neural dynamics favor different spatial granularities. Overall, these results demonstrate that while the model is sensitive to temporal granularity (t), it remains robust across a wide range of area-based configurations (A, C_a), validating the effectiveness and adaptability of the proposed Area-wise Sliding Window Attention in modeling diverse neural data structures.

Table 8: Comprehensive sensitivity analysis of tokenization hyperparameters.

t	w	A	C_a	M1-CO1 (Multi-day)		MC-Maze (R^2)	Area2-Bump (R^2)
				Balanced Accuracy	Weighted F1		
2	10	8	32	0.670	0.672	0.790	0.812
5	10	8	32	0.853	0.856	0.846	0.866
10	10	8	32	**0.909**	**0.910**	**0.890**	<u>0.890</u>
20	10	8	32	0.856	0.859	0.867	0.850
10	10	8	8	0.824	0.826	0.836	0.875
10	10	8	16	0.878	0.880	0.874	0.871
10	10	8	64	0.872	0.871	0.860	0.882
10	4	2	32	0.837	0.835	0.863	**0.901**
10	6	4	32	0.861	0.862	0.885	0.880
10	18	16	32	0.848	0.853	0.874	0.882

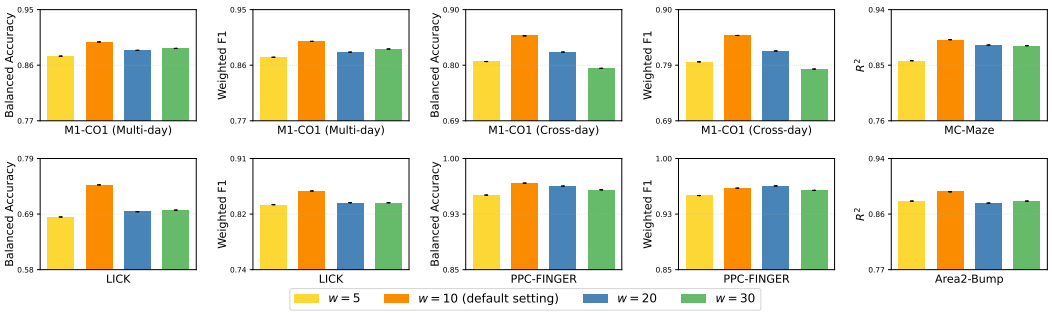

Figure 7: Sensitivity analysis of different sliding window sizes w in ASWA.

Given the dataset-dependent interaction between A and w observed above, we further isolate the effect of the sliding window size w in the Area-wise Sliding Window Attention (ASWA) mechanism by varying $w \in \{5, 10, 20, 30\}$. As shown in Figure 7, the model's performance degrades when

$w = 5$. This occurs because a window size smaller than the number of areas ($A = 8$) hinders the model's ability to establish necessary temporal dependencies across area boundaries, enabling an area in the current interval to attend to its corresponding features from the preceding interval. When w is increased to 10 as our default setting, the model achieves the best performance, as this size is sufficient to bridge the inter-interval gap. Further increasing the window size to 20 or 30 does not yield significant gains, suggesting that neural dynamics are predominantly governed by local temporal dependencies. Therefore, a compact sliding window is sufficient for robust representation learning without introducing redundant computation or long-range noise.

J VISUALIZATION

J.1 CONTEXT REPRESENTATIONS

As shown in Figure 8, we investigate the semantic consistency and discriminability of text representations to better understand the role of natural language as a modality alignment bridge in cross-dataset and cross-species neural pretraining. A key challenge in building unified feature spaces for heterogeneous neural signals, such as spike trains recorded from different brain regions, tasks, and species, lies in achieving robust alignment across highly non-stationary experimental conditions. To address this, we introduce structured natural language descriptions as a shared semantic interface and analyze their embedding properties using a pretrained text encoder such as MiniLM-L6-v2 (Wang et al., 2021). Specifically, we construct structured prompts based on dataset metadata from both pretraining datasets (M1-CO1, M1-CO2, Pac-Man, HPC-HG, LICK) and downstream tasks, following the template:

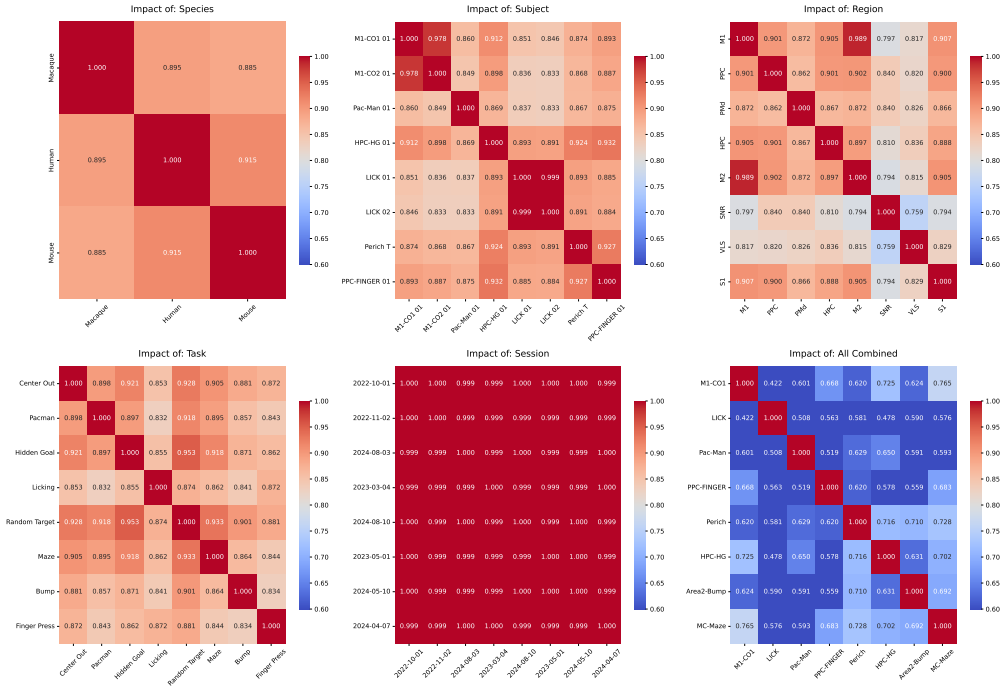

Figure 8: Semantic consistency and discriminability of text representations. Heatmaps show cosine similarities under controlled variable settings, where one attribute (species, subject, region, task, or session) is varied while others are fixed, revealing strong invariance to subject and session and clear discriminative structure for task and region.

"Invasive spike signals of [Species] species ([Dataset] [Subject]) in the [Region] brain region during the [Task] task under session [Session/Date]"

We then apply a controlled variable strategy, where we fix all but one factor and systematically vary a single attribute, including species, subject, brain region, task, and session date. Cosine similarities

between corresponding text embeddings are computed to generate heatmaps that visualize semantic variation across different dimensions. The results show strong semantic stability with respect to session and subject variations, where similarity values remain consistently above 0.99, indicating that the text encoder effectively smooths inter-day and inter-subject variability and captures invariant experimental structure. In contrast, task and brain region dimensions exhibit clear discriminative gradients, with similarities ranging between 0.85 and 0.95, suggesting that the text encoder can effectively distinguish functional differences such as M1 versus hippocampus or center-out versus Pacman tasks, thereby providing informative task-level guidance. Furthermore, when multiple factors are jointly varied, the similarity decreases, in some cases below 0.6, demonstrating strong global differentiation capability. Overall, these results indicate that structured textual descriptions preserve high semantic consistency while still maintaining sufficient discriminability to distinguish heterogeneous neural datasets, thereby validating natural language as an effective and generalizable bridge for cross-dataset neural representation learning and knowledge transfer.

J.2 Spike Token Representations

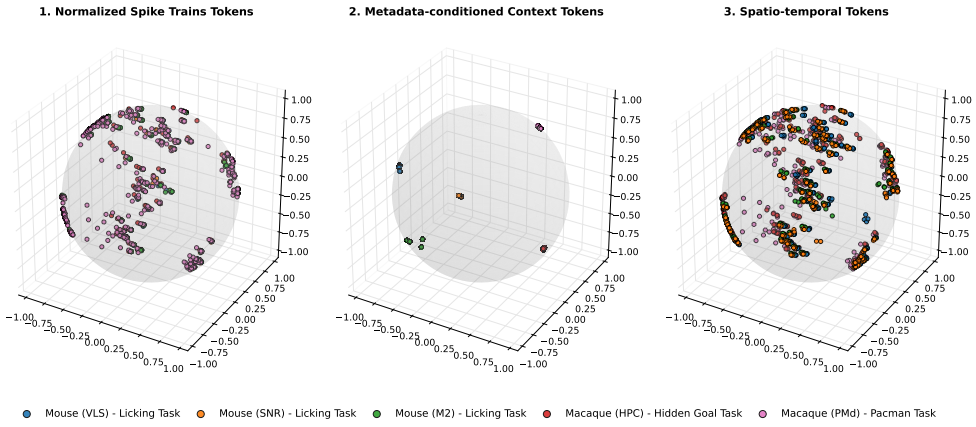

Figure 9: Token distributions on a hyperspherical space across stages, showing how metadata progressively structures neural representations from entangled spike tokens to organized spatiotemporal embeddings.

As illustrated in Fig. 9, we visualize token representations at different stages using hyperspherical embedding projections to examine how metadata injection reshapes the feature space for cross-species and cross-task neural signals. When tokens are derived solely from raw spike trains, their embeddings are highly entangled and uniformly dispersed on the hypersphere, with substantial overlap across species such as mouse and macaque as well as different brain regions, indicating limited ability to capture domain-specific structure and weak robustness under domain shift. By introducing structured metadata, including species, subject identity, brain region, and task, as context embeddings, the model constructs a more organized representation space where tokens are distributed around distinct contextual anchors, forming separable clusters that reflect underlying experimental conditions and provide a semantic prior for neural signal modeling. After integrating contextual information, the resulting spatiotemporal tokens exhibit a more structured distribution characterized by both global coverage and local compactness, with improved uniformity on the hypersphere while preserving meaningful intra-domain variation, suggesting that metadata acts as a conditioning signal that regularizes representation learning and enhances cross-domain generalization. Overall, metadata injection transforms neural representations from entangled, low-discriminative embeddings into a structured semantic space on the hypersphere, enabling a shift from purely data-driven fitting toward context-aware representation alignment.

J.3 Encoder Representation

To qualitatively evaluate the discriminative capability of the learned neural representations, we conduct a t-SNE visualization on the M1-CO1 macaque dataset, which consists of 232-channel spike recordings from the M1 cortex. We select data from two unseen days comprising six independent

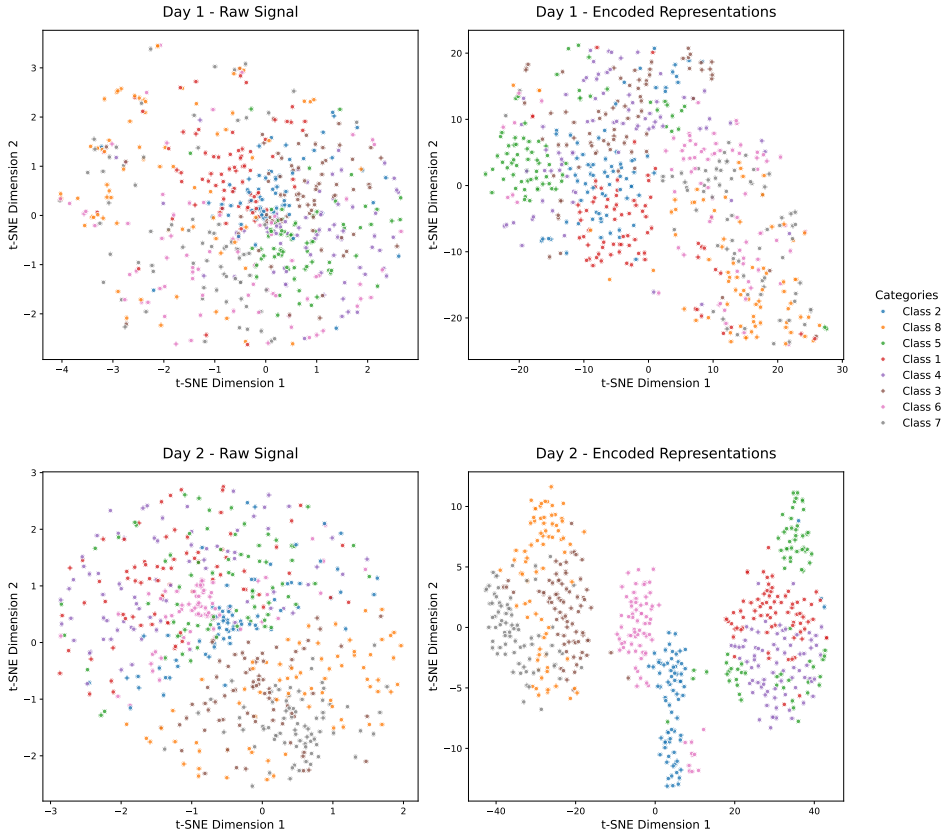

Figure 10: t-SNE visualization of feature representations on M1-CO1, comparing raw neural signals (left) and encoded representations (right).

sessions that are not included during pretraining, and consider an eight-class center-out movement classification task. As illustrated in Figure 10, we compare the distribution of raw neural signals with the learned representations produced by our model. In the raw signal space, samples from different movement directions are highly entangled across both Day 1 and Day 2, exhibiting no clear class boundaries and overlap among categories. In contrast, in the learned representation space, the encoded features form more compact and well-separated clusters, with clear distinctions between different classes. This result demonstrates that, despite not being exposed to these sessions during pretraining, the proposed model is able to extract discriminative patterns from high-dimensional, sparse, and noisy neural spike sequences that are strongly correlated with movement direction.

K LIMITATIONS

Despite the strong empirical performance and generalization ability demonstrated across diverse datasets, several limitations of the current framework should be acknowledged. First, the model relies on pre-defined spike train normalization (e.g., temporal binning and fixed channel grouping) and a structured tokenization pipeline, which impose spatial assumptions on data representation. While this design enables efficient unification of heterogeneous datasets, it may restrict flexibility when handling more complex or irregular recordings, such as variable-density electrode layouts or continuous-time neural events. Second, although UniBCI incorporates contextual metadata embeddings, it remains fundamentally limited to spike-based modalities and does not explicitly support multimodal neural data integration. This constrains its ability to leverage complementary information from other sources, such as neuroimaging (e.g., fMRI, calcium imaging) or behavioral streams, which are increasingly important for comprehensive brain modeling. Future work will therefore focus on developing adaptive, data-driven tokenization schemes and extending the framework toward unified multimodal representation learning.