Neural Data Transformer 2: Multi-context Pretraining for Neural Spiking Activity

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

The neural population spiking activity recorded by intracortical brain-computer interfaces (iBCIs) contain rich structure. Current models of such spiking activity are largely prepared for individual experimental contexts, restricting data volume to that collectable within a single session and limiting the effectiveness of deep network models. The purported challenge in aggregating neural spiking data is the pervasiveness of context-dependent distribution shifts. However, large scale unsupervised pretraining by nature spans heterogenous data, and has proven a fundamental recipe for successful representation learning across deep learning. We thus develop Neural Data Transformer 2 (NDT2), a spatiotemporal Transformer for neural spiking activity, and demonstrate pretraining can leverage motor BCI datasets that span sessions, subjects, and experimental tasks. NDT2 enables rapid adaptation to novel contexts in downstream decoding tasks, and opens the path to deployment of pretrained DNNs for iBCI control. Code will be released with publication and available for reviewers in supplementary materials.

1 Introduction

Intracortical neural spiking activity contains rich statistical structure reflecting the processing it subserves. For example, motor cortical activity during reaching is characterized with low-D dynamical models [1, 2], and these models can predict behavior under external perturbation and provides an interpretive lens for motor learning [3–5]. However, these models are currently prepared per experimental context, meaning separate datasets are collected for each cortical phenomena in each subject, for each session. Meanwhile, spiking activity structure is at least somewhat stable across these contexts; for example, dominant principal components (PCs) of neural activity can remain stable across sessions, subjects, and behavioral tasks [6–9]. This structure persists in spite of turnover in recorded neurons, physiological changes in the subject, or task changes required by the experiment [10, 11]. Conserved neural population structure suggests the opportunity for models that span beyond single experimental contexts, enabling more efficient, potent analysis and application.

In this work we focus on one primary use case: neuroprosthetics powered by intracortical brain-computer interfaces (iBCIs). With electrical recordings of just dozens to hundreds of channels of neuronal population spiking activity, today's iBCIs can relate this observed neural activity to behavioral intent, achieving impressive milestones such as high speed speech decoding [12] and high degree of freedom control of robotic arms [13]. Even so, these iBCIs currently require arduous supervised calibration in which neural activity on that day is mapped to behavioral intent. At best, cutting-edge decoders have included training data from across several days, producing thousands of trials but still modest by deep learning standards [12]. Single-session models still dominate the Neural Latents Benchmarks (NLB), a primary representation learning benchmark for spiking activity [14].

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

Thus, despite the scientifically observed conserved manifold structure, there has been little adoption of neural population models that can productively aggregate data from broader contexts.

One possible path forward is deep learning's seemingly robust recipe for leveraging heterogeneous data across domains: a generic model backbone (e.g. a Transformer [15]), unsupervised pretraining over broad data, and lightweight adaptation for a target context. The iBCI community has set the stage for this effort, for example with iBCI dataset releases (Section A.1) and NDT1 [16], which shows Transformers only need modest changes to apply to spiking activity (at least in single session datasets). We hereafter refer to NDT as NDT1. Building on this momentum, we report that Transformer pretraining can apply to motor cortical neural spiking activity from iBCIs, and allows productive aggregation of data across contexts.

Contributions: We contribute NDT2, a Transformer that pretrains over broad data sources of motor cortical spiking activity. NDT2 modifies NDT1 to improve scaling across heterogeneous contexts in 3 ways: spatiotemporal attention, learned context embeddings, and asymmetric encode-decode [17]. We find positive transfer with data from different data sessions, subjects, and tasks, and quantify their relative value. Once pretrained, NDT2 can be rapidly tuned in novel experimental sessions. We focus on offline evaluation on motor applications, demonstrating NDT2's value in decoding unstructured monkey reaching and human iBCI cursor intent.

1.1 Related Work

Unsupervised neural data pretraining. Unsupervised pretraining is particularly appealing in neuroscience due to limited data availability for most supervised tasks. We compare some of the pretrained models in different neural data modalities in Table 1. There is a remarkable convergence in modeling design despite modality diversity: 3 of 4 neural approaches use masked autoencoding, and 3 of 4 use a Transformer backbone. However, pretraining in each modality poses different challenges. Pertinent for spiking activity is the issue of data instability. While the fine spatial resolution of iBCI microelectrode arrays provide the signal needed for high-performance rehabilitation applications, it also causes high sensitivity to shifts in recording conditions. iBCIs typically require recalibration within hours, relative to ECoG-BCIs that may not require recalibration for days [18]. At the macroscopic end, EEG and fMRI can mostly address inter-measurement misalignment through preprocessing (e.g. registration to an atlas).

Table 1. Neural data pretraining. NDT2, like contemporary neural data models, aim for BERT-scale [19] pretraining. Neural models vary greatly in task quality and data encoding; invasive methods severely restrict subject count available (especially with public data). Volume is estimated as full dataset size / model input size.

Modality	Task	Estimated Pretraining Volume	Subjects
Spikes (NDT2)	Motor reaching	0.25M trials	$\sim$12
SEEG: LFP [20]	Movie Viewing	3.2M trials / 4.5K electrode-hours	10
ECoG: LFP [21]	Naturalistic behavior	0.04M trials / 108 days [22]	12
EEG [23]	Clinical assessment	0.5M trials / (26K runs [24])	11K
fMRI [25]	Varied (34 datasets)	1.8M trials (12K scans)	1.7K
BERT [19]	Natural Language	1M 'trials' (3.3B tokens)	-

Data aggregation for iBCI. Multi-context data aggregation for iBCI has largely been limited to multi-session aggregation, and is moreover typically studied in highly structured tasks. Within this scope, data are often combined through a method called stitching [26]. For context, spiking events recorded on microelectrode arrays are sometimes "sorted" according to their electrical waveforms, attributing them to putative neural units. Such a sorting process produces inherently inconsistent data dimensions across sessions, but as mentioned, activity across sessions has been observed to share consistent subspace structure, as e.g. identified by PCA. Thus, the stitching strategy aims to extract this stable subspace (and also resolve neuron count differences) by learning readin and readout layers per session. Stitching is regularly applied for BCI applications over half a year [27–29, 11]. However, learnt layers incurs parameters proportional to model size and neuron count (e.g. $128^2 = 10K$ params), which may be costly in clinical iBCI settings that comprise only a few dozen trials.

Alternatively, many iBCI systems simply forgo spike sorting after observations of minor performance gains [10, 30]. Then, input dimensions are constant across sessions, and multi-session data can feed directly into a single model [10, 31, 32] (even if the units recorded in those dimensions shift [33]).

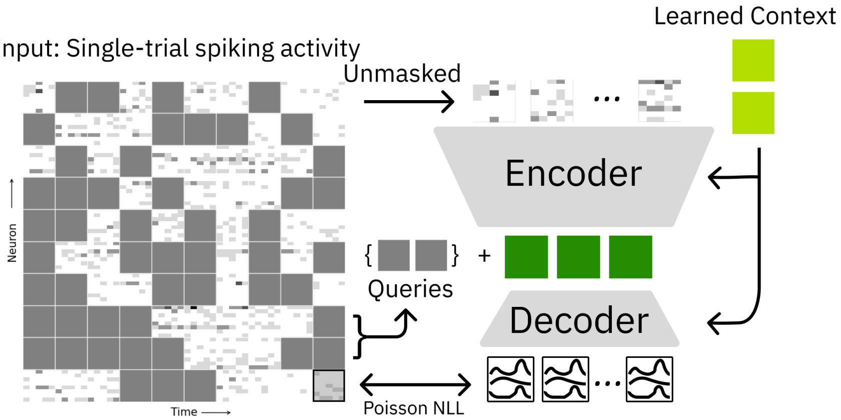

Figure 1. NDT2 is a spatiotemporal Transformer encoder-decoder. In pretraining, a spike rate decoder performs masked spike reconstruction; downstream, additional decoders directly use the encoded representations. Both the encoder and decoder share learned context embeddings representing known metadata, such as subject identity.

Note that these referenced models also typically incorporate augmentation strategies centered around channel gain modulation, noising, or dropout, emphasizing robustness as a design goal.

Domain-adaptive vs. domain-robust decoding. BCI decoder context-robustness can be explicitly promoted by either aligning data recorded in novel contexts to those of a known context, or by building decoders that are robust to context changes. Adaptive approaches realign novel contexts by learning an input mapping that minimizes distributional distance of input encodings explicitly [11, 34, 35]. Robust approaches aim to learn decoders that are agnostic to variability in recorded populations, by promoting invariant representations through model or objective design [32, 36, 37].

While robustness to context changes are a sensible functional goal for BCI *systems*, we need not force this robustness on our neural data decoders. That is, we can build adaptation into model design and evaluation. A full BCI software and hardware ecosystem can provide additional information in a way that does not impair usage. This can for example be as simple as allowing test time tuning. Such calibration in novel contexts need not be expensive: e.g. it may be passively collected unsupervised data, or freely collected metadata like subject identity. Outside of neuroscience, thin adaptation mechanisms enable robotic policies to operate in novel environments [38, 39], and language models to flexibly perform many tasks [40–42].

2 Approach

2.1 Designing Transformers for unsupervised scaling on neural data

Transformers prepared with masked autoencoding are a competitive model for representation learning on spiking neural activity in single contexts, as measured by their performance on the NLB [14]. Cross-domain resources further provide extensive technical infrastructure and relatively charted scaling properties. Thus, we retain the core model recipe, focusing instead on input design.

iBCI spiking activity is spatiotemporal. However, unlike the many pixels in vision domains, the few hundred neurons in neuronal “space” is small enough to not computationally require compression. NDT1 [16] thus only attended across space. Yet the meaning of individual neurons change across contexts, so spatial compression and attention may confer statistical benefits. STNDT [43] and EIT [37], adopt, for example, factorized spacetime attention. The former provided favorable single-session performance on the NLB, while the latter demonstrated improved multisession transfer. More generally, since factorization can impair performance [44], we might consider full spacetime attention over individual neuronal units.

Of course, scaled pretraining must weigh any potential benefits against computational efficiency. Yet while full attention is more expensive, factorizing has the subtler cost of padding overhead from data heterogeneity in either space or time, as opposed to full attention’s cost in data “area”. In pilot experiments we find comparable performance at convergence and thus focus on a full spacetime implementation. This choice also enables easy adoption of the asymmetric encoder-decoder proposed in [17], which provides memory savings by only introducing the masked proxy tokens the model aims to reconstruct in a thin decoder. We next consider resolution. In time, iBCI applications benefit from control rates of 50-100Hz [45]; we adopt 50Hz (20ms bins). In space, at present, 100-200 channels are used, but future devices are likely to record thousands of channels at a time. With context budgets of e.g. 2000 tokens, we cannot afford individual channel spatial processing. Like in ViTs [46], we propose using K-neuron patches, padding data to the nearest multiple of K. The patch is embedded by concatenating its constituent spike count embeddings, which are learned.

We also provide learned context embeddings (i.e. more tokens) to NDT2 encoder and decoders. This mechanism enables cheap model specialization given known context metadata, analogous either to prompt tuning [40] or environment embeddings [38]. We factorize context embeddings into task, subject, and session embeddings.

2.2 Datasets

We pretrain models over an aggregation of datasets (see Section A.1). All data contains single-(sorted) or multi-unit (unsorted) spiking activity recorded from either monkey or human primary motor cortex (M1) during motor tasks. We bin activity at 20ms as appropriate for motor BCI. In particular, we focus evaluation on a publically available monkey dataset, where the subjects performed self-paced reaching to random targets generated on a 2D screen (Random Target Task, RTT) [47], and unpublished human clinical BCI datasets. RTT contains both sorted and unsorted activity from 2 monkeys over 47 sessions ($\sim$20K seconds per monkey) and is suited for evaluating scaling: it contains several long sessions (near 1h), and the task is relatively challenging — decoding performance steadily improves with more data (within the same session) [48]. For comparison, in another NLB task that uses cued preparation and movement periods (Maze), decoding performance saturates by 500 trials [14]. Since RTT is continuous, we split each session into 1s trials.

We also study M1 activity in 2 human participants with spinal cord injury (P2 and P3). These participants have limited motor function but can modulate their cortical activity using attempted movements to control BCIs; we restrict our study to settings of 2D cursor control to be most analogous to RTT, which also restricts targets to a 2D workspace. All experiments conducted with humans were performed under an approved Investigational Device Exemption from the FDA, were approved by the university Institutional Review Board and the clinical trial is registered at clinicaltrials.gov . Informed consent was obtained before any experimental procedures were conducted. University and trial ID will be provided with unblinding. Details on the implants and clinical trial are described in [49, 13].

3 Results

We demonstrate the three requirements of a pretrained spiking neural data model for BCI: 1) an effective architecture, 2) beneficial scaled pretraining, and 3) practical deployment.

Model preparation and evaluation. Most initial experiments use a 6-layer, 256 hidden size encoder ($\sim$3M parameters), similar to settings in the NDT1 codebase. NDT2 uses a 2-layer decoder (0.7M parameters); we run controls to ensure this extra capacity does not benefit comparison models. To ensure that our models are not bottlenecked by compute or capacity, models are trained to convergence - with early stopping - and progressively larger models were trained until no return was observed. We pretrain with causal attention, as online iBCI decoding must be causal (though bidirectional attention improves modeling). We pretrain with 50% masking and dropout of 0.1. Further hyperparameters are not swept in general experiments; initial settings were manually tuned in pilot experiments and verified to be competitive against hyperparameter sweeps. Further training details are in Section A.2. We briefly compare against prior reported results, but to our knowledge there is no other work that attempts similar pretraining, so we primarily compare within NDT-family design choices.

We evaluate models on randomly drawn held-out test data from select "target" sessions (selection is specified per experiment). Models calibrate to these sessions with the remaining data unless specified, typically through fine-tuning, or sometimes in pretraining. We observed no differences. As unsupervised evaluation, we simply use the Poisson negative log-likelihood (NLL) objective, i.e. the reconstruction of randomly masked bins of test trials. As supervised evaluation, we report R^2 of decoded kinematics, i.e. a 2D velocity of the reaching effector. Note that while we find joint tuning

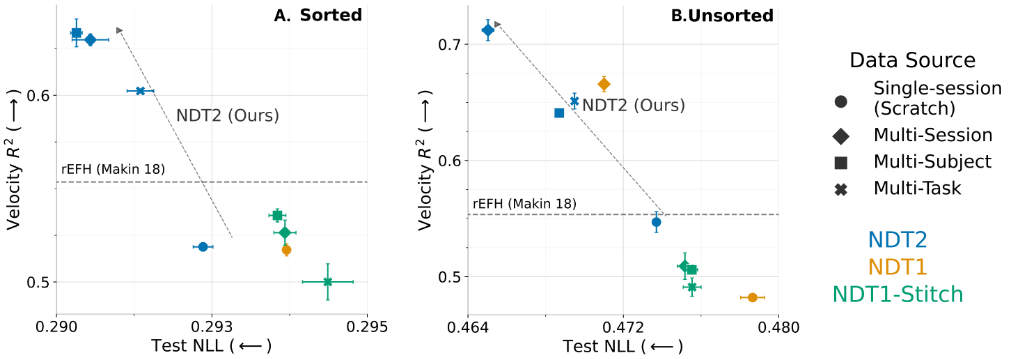

Figure 2. Pretraining architectures compared.. We show unsupervised and supervised performance (average metric on 5 sessions, standard error intervals of 3 seeds) on sorted (left) and unsorted (right) spiking activity. Higher is better for R^2, lower is better for test negative log-likelihood (NLL). Data source lists pretraining distribution (size-matched around 20Ks, except scratch single-session data). NDT2 improves with pretraining with all data sources, whereas stitching is ineffective. NDT1 aggregation is helpful but does not apply beyond session transfer. A reference well-tuned decoding score from the rEFH model is estimated [48].

with both objectives provides helpful regularization for the kinematic decoder in some experiments, the supervised metric is evaluated in a separate forward pass where no spikes are masked.

3.1 NDT2 enables multicontext pretraining

We evaluate on 5 temporally spaced evaluation sessions of monkey Indy in the RTT dataset, with both sorted and unsorted processing. Both versions are important; sorted datasets discard minimal information about spike identity and are broadly used in neuroscientific analysis while unsorted datasets are frequently more practical in BCI applications. Single-context models are trained from scratch, and to match this, pretrained models are tuned separately per evaluation session. Velocity decoding is done by tuning all models further with a lightweight behavioral probe. This separate preparation controls for the decoding gains given by the MAE pretraining objective itself, rather than broader data [50, 14]. Here we provide models 5 minutes (300 training trials) for *each evaluation session*. This quantity is a good litmus test for transfer as it is sufficient to fit reasonable single-session models but also near the high end for practical calibration. A 10% test split is used in each evaluation session (this small % is due to several sessions not containing much more than 300 trials). We pretrain models using approximately 20K trials of data, either with the remaining non-evaluation sessions of monkey Indy (Multi-Session), the sessions from the other monkey (Multi-Subject), or from other datasets entirely (Multi-Task).

Prior work in multi-session aggregation either use stitching layers or directly train on multi-day data with consistent unit count. Thus we use NDT1 with stitching as a baseline for sorted data, and with or without stitching for unsorted data. NDT2 pads observed neurons in any dataset to the nearest patch multiple. Since we evaluate on the RTT dataset which lacks clear behavioral conditions, the stitch layers cannot be initialized with principal components regression [27]. All models identically receive context tokens.

We show the performance of these pretrained models for sorted and unsorted data in Fig. 2. For context, we show single-session performance achieved by NDT1 and NDT2, and the reported decoding performance of the nonlinear rEFH model released with the dataset [48]. This rEFH model was prepared slightly differently: its data splits are sequential and contiguous in time, whereas we use random draws in keeping with NLB. [1] Single session performance for NDT1 and NDT2 is below this baseline. (However, consistent with previous findings on the advantage of spatial modeling [43], we find single-session NDT2 provides some NLL gain over NDT1). Underperforming this established baseline is not too unexpected: NDT's performance can vary widely depending on extent of tuning (Transformers span a wide performance range on the NLB, see also Section A.2). Indeed, pretraining is valuable in part for greatly simplifying the hyperparameter tuning needed for model preparation [51].

---

[1] We estimate rEFH 20ms performance by linearly interpolating 16ms and 32ms scores reported in [48].

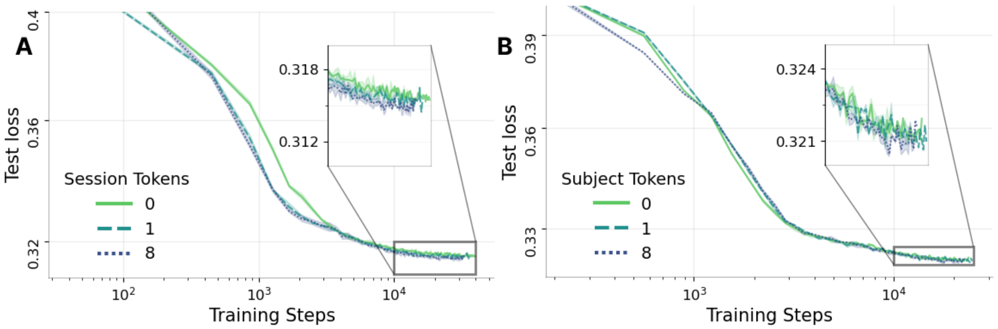

Figure 3. Context embedding ablations. **A**. Multi-session model training curves, with varying session context token count (3 seeds). Learning is improved, but additional tokens do not have notable effect. The converged score in only modestly affected. **B**. Subject transfer training curves with similarly varied token budget for subject embedding. Models receive 1 session token. There is no clear additional improvement nor harm.

However, all pretrained NDT2 models outperform these baselines, both in NLL and kinematic decoding. Surprisingly, subject-transfer works as well as session-transfer, and task-transfer provides an appreciable improvement as well. Stitching performs much worse in all cases, and in fact, task transfer brings NDT-Stitch below the single-session baseline. We expect that the underwhelming benefits of stitching is due to the lack of structure in the task.

In the unsorted case, one expects that the consistent dimensionality would particularly benefit inter-session transfer. Indeed, unsorted cross-session transfer achieves the best decoding ($> 0.7R^2$) in these experiments. Cross-task and subject decoding also improve slightly, indicating a minor benefit of unsorted decoding overall. Given this, we maintain unsorted formats in subsequent analysis of RTT. Otherwise, relative trends are consistent with the sorted case. Both analyses indicate different pretraining distributions all provide some benefit for modeling a new target context, but suggest differences e.g. between session transfer and the others. We return to a deeper comparison in Section 3.2.

Design choices. NDT2 introduces two primary design elements: context tokens, and patch size. We show the empirical optimality of 32 neuron patches in Section A.3.2; here we report on the effect of context tokens. NDT2 integrates context tokens directly by adding learned tokens and adding them to the data token sequence (i.e. in-context aggregation). Our pilots found no difference using cross-attention integration. The training curves of sorted multisession models augmented with context tokens, shown in Fig. 3A, demonstrate a primary effect in speeding convergence, which can be valuable in large scale pretraining. The benefit to converged NLL (some $1e-3$) is modest but non-negligible, considering the NLL resolution in Fig. 2. This trend replicates at smaller data scales (Fig. 10). Providing 1 session token and additionally varying the available subject tokens (Fig. 3B) has much smaller effects. However, given no visible harm and negligible compute overhead, we hold as a default policy to provide 1 token for each of session, subject, and task. We revisit the supervised benefits in Section 3.3.

3.2 NDT2 scaling across contexts

Given an architecture that can aggregate contexts, a natural goal is to identify what data can be productively aggregated. For example, the extreme of pretraining over spiking activity from all brain areas in a single model is likely unproductive given how sparsely we sample the full range of neural activity. To inform future scaling efforts, we perform three analyses to coarsely estimate the transfer affinities [52] of the three delineated context classes (cross-session, subject, and task). Previously these relationships have been grounded in shared linear subspaces [6–8]; we now quantify this in the more general generative model encompassed by DNN performance transfer.

Scaling pretraining size. In Fig. 4A,B, we consider both unsupervised and supervised transfer as we scale pretraining size, given 100 trials of calibration in a novel context. The in-distribution skyline is given by the scaling of intra-session trials. First, there is a practical degree of positive transfer. At the extreme, the largest cross-session model tuned with 100 trials is comparable to a 1000-trial intra-session model. This indicates capture of a considerably long tail of neural variance (experiments

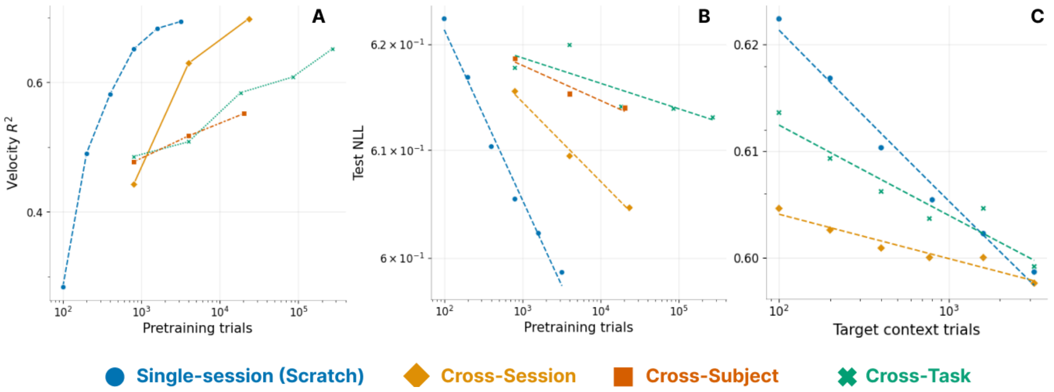

Figure 4. Scaling of transfer on RTT. We compare supervised R^2 (**A**) and unsupervised NLL scaling (**B**) as we increase the pretraining dataset size. Each point is a model that has been calibrated with 100 trials of evaluation session data. All pretraining improves on training from a single-session from-scratch model, but the benefit varies by source distribution. **C**. We seek a convergence point between pretraining and training from scratch, as we increase the number of trials we use in our target context. Models converge by 3K trials.

are rarely much larger than 1000 trials). This transfer is analogous, for example, to retained overlap in the first K PCs across two sessions, but generalizes nonlinearly and to many more sessions.

However, the shallower slopes for all other modalities indicate poorer transfer. In the unsupervised case (Fig. 4A), cross subject and task transfer never exceed the NLL achieved with 400 single-session trials. Even with an unrealistically extrapolated constant slope, we would need several orders more data before surpassing already feasible unsupervised modeling. Note our task scaling may be pessimistic as we mix human data (Table 2) with monkey data to prepare the largest model, but the trend before this point is still shallow. Interestingly, however, these limitations do not clearly translate to the supervised deployment, mirroring [53]. For example, the decode R^2 achieved by the largest model in each modality is more competitive with in-session scaling than the same comparison in NLL (far right, Fig. 4B vs A).

Convergence point with from-scratch models. We study the returns from pretraining as we vary target context calibration sizes [54]. Both models yield returns up to 3K trials, which represents about 50m of data collection in the monkey datasets, and coincidentally is the size of the largest dataset in [47]. Session transfer is again ideal, but task transfer also allows a halving of the experimental budget to achieve the same unsupervised performance. This indicates pretraining is reasonably complementary to scaling target session collection efforts. This need not have been the case: even Fig. 4B suggests that task transfer by itself is ineffective at modeling the long tail of neural variance. Note that returns on supervised evaluation are likely similar or better based on Fig. 4A/B; we explore a related idea in Section 3.3.

Overall, the returns on using pretrained BCI models depends on the use case. If we are interested in best explaining neural variance, pretraining alone underperforms a moderately large in-day data collection effort (scratch trace achieves lowest NLL in Fig. 4B). However, we do not see interference [54] in our experiments, where pretraining then tuning underperforms a from-scratch model. Thus, so long as we can afford the compute, broad pretraining is advantageous; we show these trends are repeated for two other evaluation sessions in Section A.5. We reiterate that our pretraining effort is modestly scaled; the largest pretraining only has 2 orders more data than the largest intra-context models. These conclusions may further strengthen insofar if we are able to better scale curation of pretraining data over individual experimental sessions.

3.3 Pretraining for improved decoding on novel days

RTT Decoding. In current BCI deployment, we assume the best case scenario, having both broad unsupervised data but also multiple sessions worth of supervision for our decoder. Thus, we can follow the 1st stage unsupervised pretraining with a 2nd stage of supervised pretraining of a decoder, and finally measure the decoding performance in a novel target session in Fig. 5. We find that given *either* supervised or unsupervised calibration (Sup tune, Unsup tune) in our target session, we achieve decoding performance on par with the best from-scratch models. This is true both in the realistic case where the majority of target-session data are unlabeled (Scratch - 100 Trial Sup), and with the most optimistic scenario when thousands of trials of supervised data are available. As expected, pretrained decoders provide greater gains when target session data are limited. We also find that session-adaptation is valuable, as a decoder which does not use context, while deployable without any calibration, cannot achieve the same performance. In sum, pretraining allows a degree of calibration without explicit enforcement of domain adaptation (as explored e.g. in [11, 34, 35]).

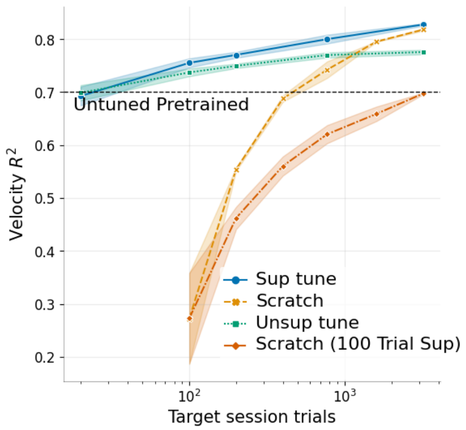

Figure 5. Tuning adaptive, pretrained decoders. Pretrained decoders with novel day calibration, whether supervised or not, outperform from-scratch models and untuned models that do not encode context with as few as 20s of data. Standard error shown with 3 seeds.

Human BCI evaluation We run a similar analysis of data transfer in offline decoding of human motor intent. Pretraining data now comprises attempted or BCI-based 2D cursor control; this is a substantial and challenging shift from actual reach in monkeys. Individual sessions contain low trial counts (e.g. 40), and velocity intent labels are much noisier than movement recordings (intent label creation is described in Section A.1). We now reserve a temporally contiguous experimental block for evaluation, but only tune one model over this block, rather than per session, due to the high session count. We also increase test split to 50% to decrease evaluation noise from low trial count. Results are shown in Table 2. We first compare broader pretraining against the cross-session regime available to a given subject (∼1-3 days of experiment time). Consistent with the previous efficacy of cross-session transfer, we see very minor improvements gained by sharing data across participants (row 1 vs 2). In human data we have a new setting with multiple tasks performed by the same participants; pretraining across multiple task contexts aids modeling of our evaluation data (row 1 vs 3). These points of reference are the best floors as single-sessions provide far too few trials to fit a DNN, and even simple linear decoders often do poorly (row 1 vs 8).

Given reports of monkey to human transfer [55], we also assess whether monkey data in either pretraining or decoder preparation improves decoding (rows 5-8). We find that monkey data, however incorporated, reduces offline decoding performance (row 5-8 < 1). Overall, these analyses show little transfer across single human subjects; this suggests revisiting of data curation in future pretraining.

Table 2. Human reach intent decoding. We show the decoding performance for 2 subjects for several data preparations. Each row shows pretraining model beyond the base task- and subject-specific data. SEM is given across 3 fine-tuning seeds. Base data is 100K trials for P2 and 30K trials for P3. Pretraining transfers across task and somewhat across subject, but *no* benefit from monkey data.

	Neural data (Unsup. pretrain)			Behavior (Sup. pretrain)	Velocity R^2 (↑)	
	Subject	Task	+130K Monkey	+24K RTT Monkey	P2	P3
1)	✓	✓			$0.503_{\pm 0.020}$	$0.515_{\pm 0.008}$
2)		✓			$0.487_{\pm 0.007}$	$0.509_{\pm 0.016}$
3)	✓				$0.444_{\pm 0.007}$	$0.493_{\pm 0.002}$
4)	✓	✓	✓	✓	$0.486_{\pm 0.012}$	$0.472_{\pm 0.019}$
5)	✓	✓	✓		$0.490_{\pm 0.007}$	$0.477_{\pm 0.018}$
6)	✓	✓		✓	$0.474_{\pm 0.009}$	$0.491_{\pm 0.010}$
7)				✓	$0.443_{\pm 0.005}$	$0.455_{\pm 0.013}$
8)	Smoothed spike ridge regression (OLE)				0.077	0.208

4 Discussion

NDT2 is a proof of concept that broad pretraining improves modeling of motor iBCI spiking activity. With simple modifications to the masked autoencoding Transformer that has been broadly adopted across domains, NDT2 at once spans the different distribution shifts faced in spiking data. For rehabilitative BCI, NDT2's simple recipe for multisession aggregation is promising even if the ideal scenario of cross-species transfer seems unlikely. More broadly, we conclude that pretraining, even at a modest 10-100K trials, is useful in realistic deployment scenarios with varied levels of supervised data.

Limitations. NDT2 design can be refined in several ways. For example, we do not claim that full spacetime attention is necessary over factorization. While we identify positive transfer in several scenarios, more precise mapping of context affinity and transfer [52] may be valuable. Further, it is difficult to extrapolate the benefits of scaling beyond what was explored here, particularly with gains in unsupervised reconstruction appearing very limited. Our evaluation also has a limited scope: we model offline reach and cursor control, and task generality is still constrained to similar motor paradigms. However, these behaviors are more general than previous demonstrations of context transfer [11, 35, 32, 36], suggesting that this approach may have broader applications. Evaluating more complex behavior decoding is a practical priority. For example, pilot experiments with real-time decoding demonstrate that these models can be deployed successfully, but also indicate nuance in translating offline to online improvements [29]. Also, design parameters such as masking ratio may affect scaling trends, which we cannot assess due to compute limits.

Negative NLB result. NDT2 performance did not exceed current NLB SoTA on motor datasets (RTT, Maze) [56]. This could simply be due to large single-session variability (which we document in Section A.5). More concretely, our scaling analysis indicates that modest pretraining (100K trials) may be insufficient against well-tuned baselines, especially on unsupervised neural data recovery, which is how the NLB is evaluated. Moreover, the NLB RTT dataset has 1K trials - larger than the setting we evaluate in - and while the NLB Maze datasets include a 100 trial split, simple task structure may have accordingly shifted the goalpost.

Neural data foundations. Pretrained representation models in each subfield of neuroscience may bridge knowledge not only across neural data modalities but possibly also to vision and language interfaces that can help analyze neural data. This greater ecosystem will hinge on confidence in the individual models, built with open data and evolving, rigorous evaluation. For example, one technique in language decoding BCIs is to integrate language models to improve BCI usability [28, 12]. Similar motor priors will be task dependent; the center-out reach degenerates from continuous control into a classification task with a sufficient prior. We must carefully track whether performance gained from multimodal inputs is improving neural representations, or solely behavioral readouts.

Modeling an embodied brain. Pretrained neural data models have potential connections to broader embodied domains. How does modeling motor neural data differ from modeling human behavior, or reinforcement learning physiological motor tasks [57]? In the sensory domain, for example, there are nearly direct architectural parallels between dominant stimulus response predictions models such as V1T [58] and vanilla ViTs [46]. The development of methods to distill each model productively into the other is would be of great merit for the NeuroAI agenda.

Towards Continuously Deployed BCI. While we relax many constraints on our data sources, our evaluation is ultimately within experimental contexts. Extensions to naturalistic settings will be challenging. BCIs likely cannot continually calibrate in an unsupervised fashion with local neural data, since BCIs inherently operate in a changing domain. Observed neural signatures update interactively with the BCI itself, changing with local plasticity and user strategy. Robotics, which faces a similar "covariate shift" challenge, offers two paths forward. Shift can be mitigated with online supervision, through methods like DAGGER [59]. Analogously, BCI pseudo-supervision through methods like intent estimation [60, 61] will likely be critical for continuous deployment. The other paradigm of scaled offline or simulated learning to achieve broad domain coverage is less clearly translated, since we lack convincing closed-loop neural data simulators (though see [62, 60]. Either way, the relative value of calibrated neural data models vs behavioral decoders is unclear.

Broader Impacts. Pretrained DNN-driven iBCIs may yield large usability improvements. However, these DNNs may require further safeguards to ensure that decoded behaviors, especially in real-time control scenarios, operate within reasonable safety parameters. Also, pretraining will require data from many different sources, but the landscape around human neural data privacy is still developing. While subject count remains low, true deidentification remains difficult, requiring, at a minimum, consented data releases.

References

[1] Saurabh Vyas, Matthew D Golub, David Sussillo, and Krishna V Shenoy. Computation through neural population dynamics. *Annual review of neuroscience*, 43:249–275, 2020.

[2] Ege Altan, Sara A. Solla, Lee E. Miller, and Eric J. Perreault. Estimating the dimensionality of the manifold underlying multi-electrode neural recordings. *PLOS Computational Biology*, 17(11):1–23, 11 2021. doi: 10.1371/journal.pcbi.1008591. URL `https://doi.org/10.1371/journal.pcbi.1008591`.

[3] Daniel J O'Shea, Lea Duncker, Werapong Goo, Xulu Sun, Saurabh Vyas, Eric M Trautmann, Ilka Diester, Charu Ramakrishnan, Karl Deisseroth, Maneesh Sahani, et al. Direct neural perturbations reveal a dynamical mechanism for robust computation. *bioRxiv*, pages 2022–12, 2022.

[4] Patrick T Sadtler, Kristin M Quick, Matthew D Golub, Steven M Chase, Stephen I Ryu, Elizabeth C Tyler-Kabara, Byron M Yu, and Aaron P Batista. Neural constraints on learning. *Nature*, 512(7515): 423–426, 2014.

[5] Saurabh Vyas, Nir Even-Chen, Sergey D Stavisky, Stephen I Ryu, Paul Nuyujukian, and Krishna V Shenoy. Neural population dynamics underlying motor learning transfer. *Neuron*, 97(5):1177–1186, 2018.

[6] Juan A. Gallego, Matthew G. Perich, Stephanie N. Naufel, Christian Ethier, Sara A. Solla, and Lee E. Miller. Cortical population activity within a preserved neural manifold underlies multiple motor behaviors. *Nature Communications*, 9(1):4233, Oct 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-06560-z. URL `https://www.nature.com/articles/s41467-018-06560-z`.

[7] Juan A. Gallego, Matthew G. Perich, Raeed H. Chowdhury, Sara A. Solla, and Lee E. Miller. Long-term stability of cortical population dynamics underlying consistent behavior. *Nature Neuroscience*, 23(2): 260–270, Feb 2020. ISSN 1546-1726. doi: 10.1038/s41593-019-0555-4. URL `https://www.nature.com/articles/s41593-019-0555-4`.

[8] Mostafa Safaie, Joanna C. Chang, Junchol Park, Lee E. Miller, Joshua T. Dudman, Matthew G. Perich, and Juan A. Gallego. Preserved neural population dynamics across animals performing similar behaviour. Sep 2022. doi: 10.1101/2022.09.26.509498. URL `https://www.biorxiv.org/content/10.1101/2022.09.26.509498v1`.

[9] Max Dabagia, Konrad P. Kording, and Eva L. Dyer. Aligning latent representations of neural activity. *Nature Biomedical Engineering*, 7(4):337–343, Apr 2023. ISSN 2157-846X. doi: 10.1038/s41551-022-00962-7. URL `https://www.nature.com/articles/s41551-022-00962-7`.

[10] David Sussillo, Sergey D. Stavisky, Jonathan C. Kao, Stephen I. Ryu, and Krishna V. Shenoy. Making brain–machine interfaces robust to future neural variability. 7:13749, Dec 2016. ISSN 2041-1723. doi: 10.1038/ncomms13749. URL `https://www.nature.com/articles/ncomms13749`.

[11] Brianna M. Karpowicz, Yahia H. Ali, Lahiru N. Wimalasena, Andrew R. Sedler, Mohammad Reza Keshtkaran, Kevin Bodkin, Xuan Ma, Lee E. Miller, and Chethan Pandarinath. Stabilizing brain-computer interfaces through alignment of latent dynamics. Nov 2022. doi: 10.1101/2022.04.06.487388. URL `https://www.biorxiv.org/content/10.1101/2022.04.06.487388v2`.

[12] Francis Willett, Erin Kunz, Chaofei Fan, Donald Avansino, Guy Wilson, Eun Young Choi, Foram Kamdar, Leigh R. Hochberg, Shaul Druckmann, Krishna V. Shenoy, and Jaimie M. Henderson. A high-performance speech neuroprosthesis. Jan 2023. doi: 10.1101/2023.01.21.524489. URL `https://www.biorxiv.org/content/10.1101/2023.01.21.524489v1`.

[13] B Wodlinger, J E Downey, E C Tyler-Kabara, A B Schwartz, M L Boninger, and J L Collinger. Ten-dimensional anthropomorphic arm control in a human brain-machine interface: difficulties, solutions, and limitations. *Journal of Neural Engineering*, 12(1):016011, Feb 2015. ISSN 1741-2560, 1741-2552. doi: 10.1088/1741-2560/12/1/016011. URL `https://iopscience.iop.org/article/10.1088/1741-2560/12/1/016011`.

[14] Felix Pei, Joel Ye, David 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, 2022.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf`.

[16] Joel Ye and Chethan Pandarinath. Representation learning for neural population activity with neural data transformers, 2021. URL `https://doi.org/10.51628/001c.27358`.

[17] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. Dec 2021. doi: 10.48550/arXiv.2111.06377. URL `http://arxiv.org/abs/2111.06377`. arXiv:2111.06377 [cs].

[18] Daniel B. Silversmith, Reza Abiri, Nicholas F. Hardy, Nikhilesh Natraj, Adelyn Tu-Chan, Edward F. Chang, and Karunesh Ganguly. *Nature Biotechnology*, 39(3):326–335, Mar 2021. ISSN 1546-1696. doi: 10.1038/s41587-020-0662-5. URL `https://www.nature.com/articles/s41587-020-0662-5`.

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[20] Christopher Wang, Vighnesh Subramaniam, Adam Uri Yaari, Gabriel Kreiman, Boris Katz, Ignacio Cases, and Andrei Barbu. BrainBERT: Self-supervised representation learning for intracranial recordings. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=xmcYx_reUn6`.

[21] Sabera J Talukder, Jennifer J. Sun, Matthew K Leonard, Bingni W Brunton, and Yisong Yue. Deep neural imputation: A framework for recovering incomplete brain recordings. In *NeurIPS 2022 Workshop on Learning from Time Series for Health*, 2022. URL `https://openreview.net/forum?id=c9qFg8UrIcn`.

[22] Samuel M Peterson, Shiva H Singh, Benjamin Dichter, Kelvin Tan, Craig DiBartolomeo, Devapratim Theogarajan, Peter Fisher, and Josef Parvizi. Ajile12: Long-term naturalistic human intracranial neural recordings and pose. *Scientific Data*, 9(1):184, 2022. ISSN 2052-4463. doi: 10.1038/s41597-022-01280-y. URL `https://doi.org/10.1038/s41597-022-01280-y`.

[23] Demetres Kostas, Stéphane Aroca-Ouellette, and Frank Rudzicz. BENDR: Using transformers and a contrastive self-supervised learning task to learn from massive amounts of EEG data. *Frontiers in Human Neuroscience*, 15, 2021. ISSN 1662-5161. doi: 10.3389/fnhum.2021.653659. URL `https://www.frontiersin.org/articles/10.3389/fnhum.2021.653659`.

[24] Iyad Obeid and Joseph Picone. The temple university hospital EEG data corpus. *Frontiers in Neuroscience*, 10:196, 2016. ISSN 1662-453X. doi: 10.3389/fnins.2016.00196. URL `https://www.frontiersin.org/articles/10.3389/fnins.2016.00196`.

[25] Armin W. Thomas, Christopher Ré, and Russell A. Poldrack. Self-supervised learning of brain dynamics from broad neuroimaging data, 2023.

[26] Srini Turaga, Lars Buesing, Adam M Packer, Henry Dalgleish, Noah Pettit, Michael Hausser, and Jakob H Macke. Inferring neural population dynamics from multiple partial recordings of the same neural circuit. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL `https://proceedings.neurips.cc/paper_files/paper/2013/file/01386bd6d8e091c2ab4c7c7de644d37b-Paper.pdf`.

[27] Chethan Pandarinath, Daniel J. O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D. Stavisky, Jonathan C. Kao, Eric M. Trautmann, Matthew T. Kaufman, Stephen I. Ryu, Leigh R. Hochberg, Jaimie M. Henderson, Krishna V. Shenoy, L. F. Abbott, and David Sussillo. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, 15(10):805–815, Oct 2018. ISSN 1548-7105. doi: 10.1038/s41592-018-0109-9. URL `https://www.nature.com/articles/s41592-018-0109-9`.

[28] Francis R. Willett, Donald T. Avansino, Leigh R. Hochberg, Jaimie M. Henderson, and Krishna V. Shenoy. High-performance brain-to-text communication via handwriting. *Nature*, 593(7858):249–254, May 2021. ISSN 1476-4687. doi: 10.1038/s41586-021-03506-2. URL `https://www.nature.com/articles/s41586-021-03506-2`.

[29] Darrel R. Deo, Francis R. Willett, Donald T. Avansino, Leigh R. Hochberg, Jaimie M. Henderson, and Krishna V. Shenoy. Translating deep learning to neuroprosthetic control. Apr 2023. doi: 10.1101/2023.04.21.537581. URL `https://www.biorxiv.org/content/10.1101/2023.04.21.537581v1`.

[30] Chethan Pandarinath and Sliman J Bensmaia. The science and engineering behind sensitized brain-controlled bionic hands. *Physiological Reviews*, 102(2):551–604, 2022.

[31] Nur Ahmadi, Timothy G. Constandinou, and Christos-Savvas Bouganis. Robust and accurate decoding of hand kinematics from entire spiking activity using deep learning. *Journal of Neural Engineering*, 18(2): 026011, Feb 2021. ISSN 1741-2552. doi: 10.1088/1741-2552/abde8a. URL `https://iopscience.iop.org/article/10.1088/1741-2552/abde8a/meta`.

[32] Justin Jude, Matthew G. Perich, Lee E. Miller, and Matthias H. Hennig. Robust alignment of cross-session recordings of neural population activity by behaviour via unsupervised domain adaptation. Feb 2022. doi: 10.48550/arXiv.2202.06159. URL `http://arxiv.org/abs/2202.06159`. arXiv:2202.06159 [cs, q-bio].

[33] John E. Downey, Nathaniel Schwed, Steven M. Chase, Andrew B. Schwartz, and Jennifer L. Collinger. Intracortical recording stability in human brain-computer interface users. *Journal of Neural Engineering*, 15(4):046016, Aug 2018. ISSN 1741-2552. doi: 10.1088/1741-2552/aab7a0.

[34] Ali Farshchian, Juan A. Gallego, Joseph P. Cohen, Yoshua Bengio, Lee E. Miller, and Sara A. Solla. Adversarial domain adaptation for stable brain-machine interfaces. Jan 2019. URL `https://openreview.net/forum?id=Hyx6Bi0qYm`.

[35] Xuan Ma, Fabio Rizzoglio, Eric J. Perreault, Lee E. Miller, and Ann Kennedy. Using adversarial networks to extend brain computer interface decoding accuracy over time. Aug 2022. doi: 10.1101/2022.08.26.504777. URL `https://www.biorxiv.org/content/10.1101/2022.08.26.504777v1`.

[36] Justin Jude, Matthew G. Perich, Lee E. Miller, and Matthias H. Hennig. Capturing cross-session neural population variability through self-supervised identification of consistent neuron ensembles. In *Proceedings of the 1st NeurIPS Workshop on Symmetry and Geometry in Neural Representations*, page 234–257. PMLR, Feb 2023. URL `https://proceedings.mlr.press/v197/jude23a.html`.

[37] Ran Liu, Mehdi Azabou, Max Dabagia, Jingyun Xiao, and Eva L. Dyer. Seeing the forest and the tree: Building representations of both individual and collective dynamics with transformers. Oct 2022. URL `https://openreview.net/forum?id=5aZ8umizItU`.

[38] Xue Bin Peng, Erwin Coumans, Tingnan Zhang, Tsang-Wei Lee, Jie Tan, and Sergey Levine. Learning agile robotic locomotion skills by imitating animals. *arXiv preprint arXiv:2004.00784*, 2020.

[39] Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. 2021.

[40] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning, 2021.

[41] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp, 2019.

[42] 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. Feb 2022. doi: 10.48550/arXiv.2109.01652. URL `http://arxiv.org/abs/2109.01652`. arXiv:2109.01652 [cs].

[43] Trung Le and Eli Shlizerman. Stndt: Modeling neural population activity with a spatiotemporal transformer, 2022.

[44] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer, 2021.

[45] Maryam M. Shanechi, Amy L. Orsborn, Helene G. Moorman, Suraj Gowda, Siddharth Dangi, and Jose M. Carmena. Rapid control and feedback rates enhance neuroprosthetic control. *Nature Communications*, 8(1):13825, Jan 2017. ISSN 2041-1723. doi: 10.1038/ncomms13825. URL `https://www.nature.com/articles/ncomms13825`.

[46] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=YicbFdNTTy`.

[47] Joseph E. O'Doherty, Mariana M. B. Cardoso, Joseph G. Makin, and Philip N. Sabes. Nonhuman primate reaching with multichannel sensorimotor cortex electrophysiology, May 2017. URL `https://doi.org/10.5281/zenodo.788569`.

[48] Joseph G Makin, Joseph E O'Doherty, Mariana M B Cardoso, and Philip N Sabes. Superior arm-movement decoding from cortex with a new, unsupervised-learning algorithm. *Journal of Neural Engineering*, 15 (2):026010, Apr 2018. ISSN 1741-2560, 1741-2552. doi: 10.1088/1741-2552/aa9e95. URL `https://iopscience.iop.org/article/10.1088/1741-2552/aa9e95`.

[49] Jennifer L Collinger, Brian Wodlinger, John E Downey, Wei Wang, Elizabeth C Tyler-Kabara, Douglas J Weber, Angus JC McMorland, Meel Velliste, Michael L Boninger, and Andrew B Schwartz. High-performance neuroprosthetic control by an individual with tetraplegia. *The Lancet*, 381(9866):557–564, 2013.

[50] Kundan Krishna, Saurabh Garg, Jeffrey P. Bigham, and Zachary C. Lipton. Downstream datasets make surprisingly good pretraining corpora, 2022.

[51] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning, 2020.

[52] Amir Zamir, Alexander Sax, William Shen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning, 2018.

[53] Ashish Teku Vaswani, Dani Yogatama, Don Metzler, Hyung Won Chung, Jinfeng Rao, Liam B. Fedus, Mostafa Dehghani, Samira Abnar, Sharan Narang, and Yi Tay. Scale efficiently: Insights from pre-training and fine-tuning transformers. 2022.

[54] Danny Hernandez, Jared Kaplan, Tom Henighan, and Sam McCandlish. Scaling laws for transfer, 2021.

[55] Fabio Rizzoglio, Ege Altan, Xuan Ma, Kevin L. Bodkin, Brian M. Dekleva, Sara A. Solla, Ann Kennedy, and Lee E. Miller. Monkey-to-human transfer of brain-computer interface decoders. *bioRxiv*, 2022. doi: 10.1101/2022.11.12.515040. URL `https://www.biorxiv.org/content/early/2022/11/13/2022.11.12.515040`.

[56] Unknown. EvalAI leaderboard. `https://eval.ai/web/challenges/challenge-page/1256/leaderboard/3184`, 2022. Accessed on May 16, 2023.

[57] Vittorio Caggiano, Huawei Wang, Guillaume Durandau, Massimo Sartori, and Vikash Kumar. Myosuite – a contact-rich simulation suite for musculoskeletal motor control, 2022.

[58] Bryan M. Li, Isabel M. Cornacchia, Nathalie L. Rochefort, and Arno Onken. V1t: large-scale mouse v1 response prediction using a vision transformer, 2023.

[59] Stephane Ross, Geoffrey J. Gordon, and J. Andrew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning, 2011.

[60] Guy H. Wilson, Francis R. Willett, Elias A. Stein, Foram Kamdar, Donald T. Avansino, Leigh R. Hochberg, Krishna V. Shenoy, Shaul Druckmann, and Jaimie M. Henderson. Long-term unsupervised recalibration of cursor bcis. Feb 2023. doi: 10.1101/2023.02.03.527022. URL `https://www.biorxiv.org/content/10.1101/2023.02.03.527022v1`.

[61] Beata Jarosiewicz, Anish A Sarma, Jad Saab, Brian Franco, Sydney S Cash, Emad N Eskandar, and Leigh R Hochberg. Retrospectively supervised click decoder calibration for self-calibrating point-and-click brain–computer interfaces. *Journal of Physiology-Paris*, 110(4):382–391, 2016.

[62] Francis R. Willett, Daniel R. Young, Brian A. Murphy, William D. Memberg, Christine H. Blabe, Chethan Pandarinath, Sergey D. Stavisky, Paymon Rezaii, Jad Saab, Benjamin L. Walter, Jennifer A. Sweet, Jonathan P. Miller, Jaimie M. Henderson, Krishna V. Shenoy, John D. Simeral, Beata Jarosiewicz, Leigh R. Hochberg, Robert F. Kirsch, and A. Bolu Ajiboye. Principled bci decoder design and parameter selection using a feedback control model. *Scientific Reports*, 9(1):8881, Jun 2019. ISSN 2045-2322. doi: 10.1038/s41598-019-44166-7. URL `https://doi.org/10.1038/s41598-019-44166-7`.

[63] Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners, 2022.

A Supplementary Material

A.1 Dataset Preparation

We perform minimal preprocessing on datasets. For pretraining, we do *not* explicitly filter for successful trial outcome as done in most neuroscientific analyses (some datasets are released with erratic outcomes pre-filtered). Neither do we (beyond what is provided directly in datasets) filter for cross-correlated channels or low-firing neurons. We also do not z-score neuronal firing, both for simplicity and so as to not remove any potential cross-channel/session information. The one exception to this is that neurons with firing < 0.5Hz are removed in the sorted analysis of the O'Doherty RTT dataset, to reduce the number of spatial channels below 288. As some datasets report single unit activity and some report multi-unit activity, the dynamic range of the input data varied by an order of magnitude, with baseline firing rates varying between 0.1Hz to upwards of 50Hz. The authors believe additional data curation is likely to improve model quality.

In total, the max number of pretraining trials or pseudo trials was on the order of 100K trials. Each trial lasted up to 2.5s (cropped or chunked if trials were longer), and used all recorded M1 activity, and PMd activity if available.

Decoding targets were either in a standard unit or in z-scores against the dataset mean and standard deviation (not a session specific z-score). Standard units of meters/second were primarily used in most RTT analysis, except when preparing an RTT/human BCI decoder, which used respective z-scores.

Reaching datasets

- Neural Latents Benchmark motor datasets (`MC_Maze`, `MC_Maze_small`, `MC_Maze_med`, `MC_Maze_large`, `MC_RTT`): $\sim$3.7K trials.
- Churchland et al., obstacle-guided (maze) reaching, 2 monkeys, 9 sessions / $\sim$20K trials.
- Nir-Even Chen et al., delayed reaching, 2 monkeys, 12 sessions / $\sim$ 80K trials total.
- O'Doherty et al., self-paced reaching, 2 monkeys, 47 sessions / $\sim$40K seconds total.

Isometric manipulandum datasets

- Gallego-Carracedo et al., isometric center-out and hold, 2 monkeys, 12 sessions, $\sim$2.7K trials total.
- Dyer et al., 2 monkeys (same as above), 4 sessions/$\sim$750 trials total.

Human BCI datasets

- Human participant data from ongoing clinical trials. Subsetted to 2D cursor control activity, either under observation/attempted activity, partial, or full BCI control [Private]. During observation, participants observe a programmatically controlled cursors, which e.g. is performing center out at a steady pace in a trialized fashion. We take the programmatic cursor velocity and apply a boxcar filter of 500ms and use that as our velocity label.

A.2 Compute and Hyperparameter Tuning

The full, uncurated logs of all model preparation are available at https://wandb.ai/<REDACTED>.

Basic hyperparameters

1. In both pretraining and fine-tuning, we scale batch size (accumulating batches or using multi-GPU training when necessary) to be roughly proportional to full dataset so that each epoch requires 10-100 steps; we find performance is not too sensitive to batch size within an order of magnitude of this heuristic (especially in pretraining).
2. In pretraining we manually tuned LR to $5e-4$ in initial experiments and hold it fairly constant in pretraining. We swept learning rate in our hyperparameter comparisons below.
3. In pretraining, we use learning rate warm-up for 100 epochs, and decay to ϵ by 2500 epochs. This is a high threshold that is typically not reached: training converges within 100-1K

epochs for our manually tuned LR. In fine-tuning, we experimented with similar ramping schedule but settled on fixed small LR (which are typically grid-searched).

4. For RTT, we swept and found that a decoding lag of 120ms worked reasonably well. (This is similar to reports in [14]. For human BCI, we do not use decoding lag.
5. For human offline evaluation, we take the best of two evaluation hyperparameter settings: 10% or 50% masking during target-session tuning. We also report the R^2 only for times where the intent is non-zero; participants are not typically perfectly zero-intent during the majority of non-zero phases (i.e. data are noisy). We do not filter data by putative quality as measured by online performance in the experiment in which the data was collected; thus our calibration data includes several noisy, incomplete trials as well. Evaluation data are restricted to a contiguous set of sessions with non-trivial linear decoding.

Compute costs We estimate computational costs with respect to data volume, as model size is held relatively static (6-12 layers, 128-384 hidden size). Most analysis was run on SLURM clusters. Pilot realtime feasibility was assessed on an NVIDIA 1060/2060, where tuning took about 10 minutes and loop time was under 20ms.

1. Fitting datasets on the order of 1K trials typically requires 20m-1hr on 12G 1080/2080-series NVIDIA GPUs.
2. 10K-20K trial datasets require 2-8 32G-V100 hours.
3. 100K+ datasets require 72 80G-A100 hours.

A.3 NDT2 Design Notes

Architectural details. We refer readers to the codebase for full details, but note that NDT2 used pre-normalization layers but otherwise leave the Pytorch implementation of the Transformer layers untouched.

HP Sweeps.

We briefly show that NDT2 achieves higher performances than comparisons when sweeping across dropout ($[0.1, 0.4]$), weight decay ($[1e-3, 5e-2]$, and hidden size $(128, 256)$. NDT2 does have higher variance, but the main sensitivity is to dropout. We run this sweep and test evaluation in one training stage, Our base NDT2 uses dropout 0.1, hidden size 256, weight decay $1e-2$. In the code, this experiment is configured in `exp/arch/tune_hp, exp/arch/tune_hp_unsort`.

A.3.1 Mask Ratio.

We do not widely explore mask ratios due to compute constraints. In pilots throughout, we do not find that decoding is too sensitive to mask ratio (e.g. Fig. 7), but reconstruction quality is hard to compare as the inference problem depends on the masking ratio itself. The reasonable effectiveness of high mask ratios is consistent with general observations of low dimensionality and high redundancy in the code, compared to say, language [63].

A.3.2 Patch Size.

In early experiments we explored patch size but quickly found a tradeoff. Smaller patch sizes do appear to incrementally improve neural data models, but are both more expensive computationally (to train) and statistically (to learn decoders off of). We show this in Fig. 8. Note how the unsupervised NLL is similar or better with smaller patches, but decoding is dramatically worse, regardless of whether we mean pool across the population's tokens at each timestep (the default) or not. Smaller patches may be worth revisiting if we have a high amount of supervised data to learn a decoder with; this will likely be an empirical decision.

A.4 Additional exploratory experiments

Stitching design Our stitching implementation randomly intializes a linear readin and readout linear layer. For ease of implementation, we stitch at the output of the network encoder rather than the output of the decoder (the linear-exponential readout layer comes after per-context stitch layer). In

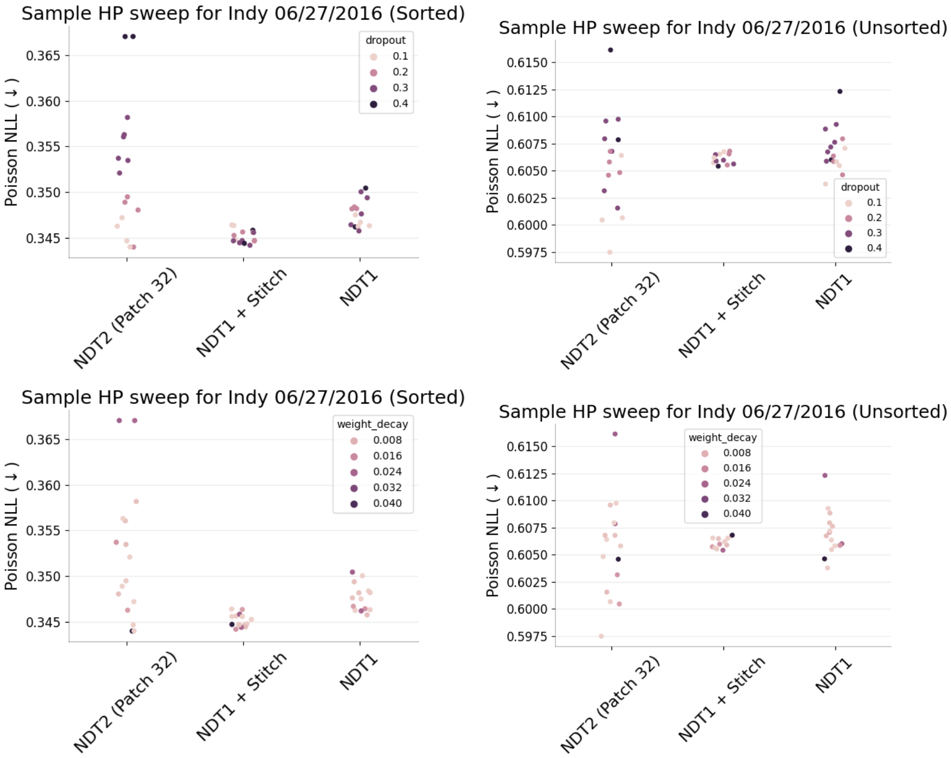

Figure 6. Sweeps for regularization parameters. NDT2 requires lower dropout.

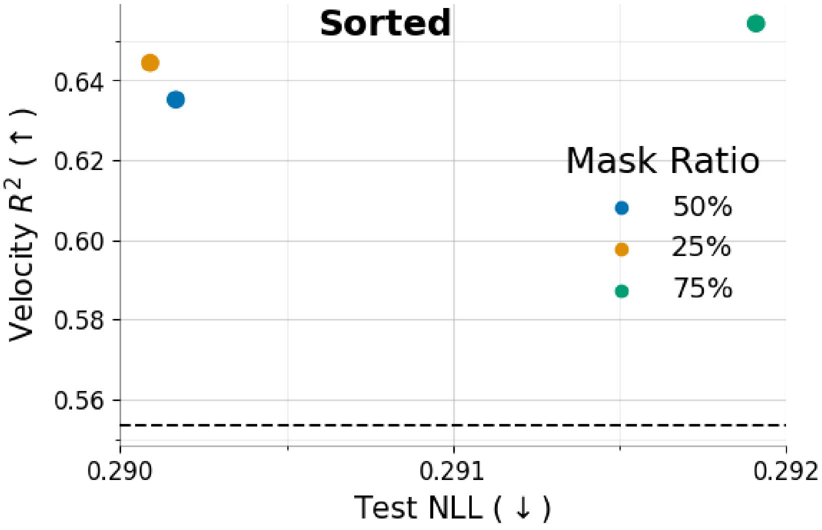

Figure 7. Mask ratios over 5 datasets. At test time, the given ratio is held out during evaluation.

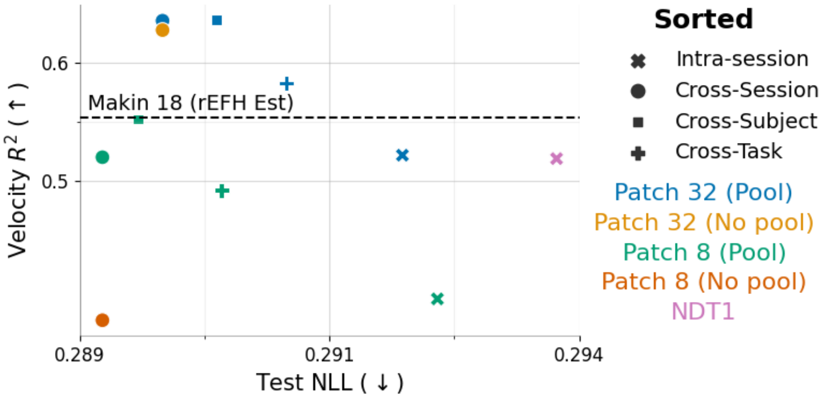

Figure 8. 32-neuron patches compared against 8-neuron patches.

unreported pilots, we find stitching into compressed dimensions (e.g. half of readin channels) to reduce the context-specific parameter count, or only including stitching at the readin or readout made no significant difference.

Kinematic decoder design There are three straightforward strategies for building a kinematic decoder with NDT2. In keeping with NLB, we could learn a linear probe on representations at each timestep, or we could use a thin Transformer decoder to allow information from multiple timesteps to aid the prediction. We experimented with both and chose the latter for minor gains. For simplicity, we run the Transformer decoder in one forward pass for all timesteps, i.e. there is no autoregressive feedback of previous kinematic estimates. We find that cross-attention for decoding queries slightly edges out in-context attention on the higher ends of the pretraining data scales we explore (e.g. 100K), and report with that setting. The primary difference is that cross-attention restricts neural data tokens from attending to the kinematic query tokens, while in-context strategies do not distinguish the two. For decoding probes, where the decoder is prepared on only a few hundred trials, we find it beneficial to mean-pool neural data tokens per timestep, and still use in-context attention (linear decoding directly works similarly).

A.5 Single-session breakdown of experimental results

Single-session variability We break open the aggregate results from our primary result figures. The primary takeaways are elaborated in each caption. Overall, we note that single datasets are insufficient to make conclusions on design choices given variability in results.

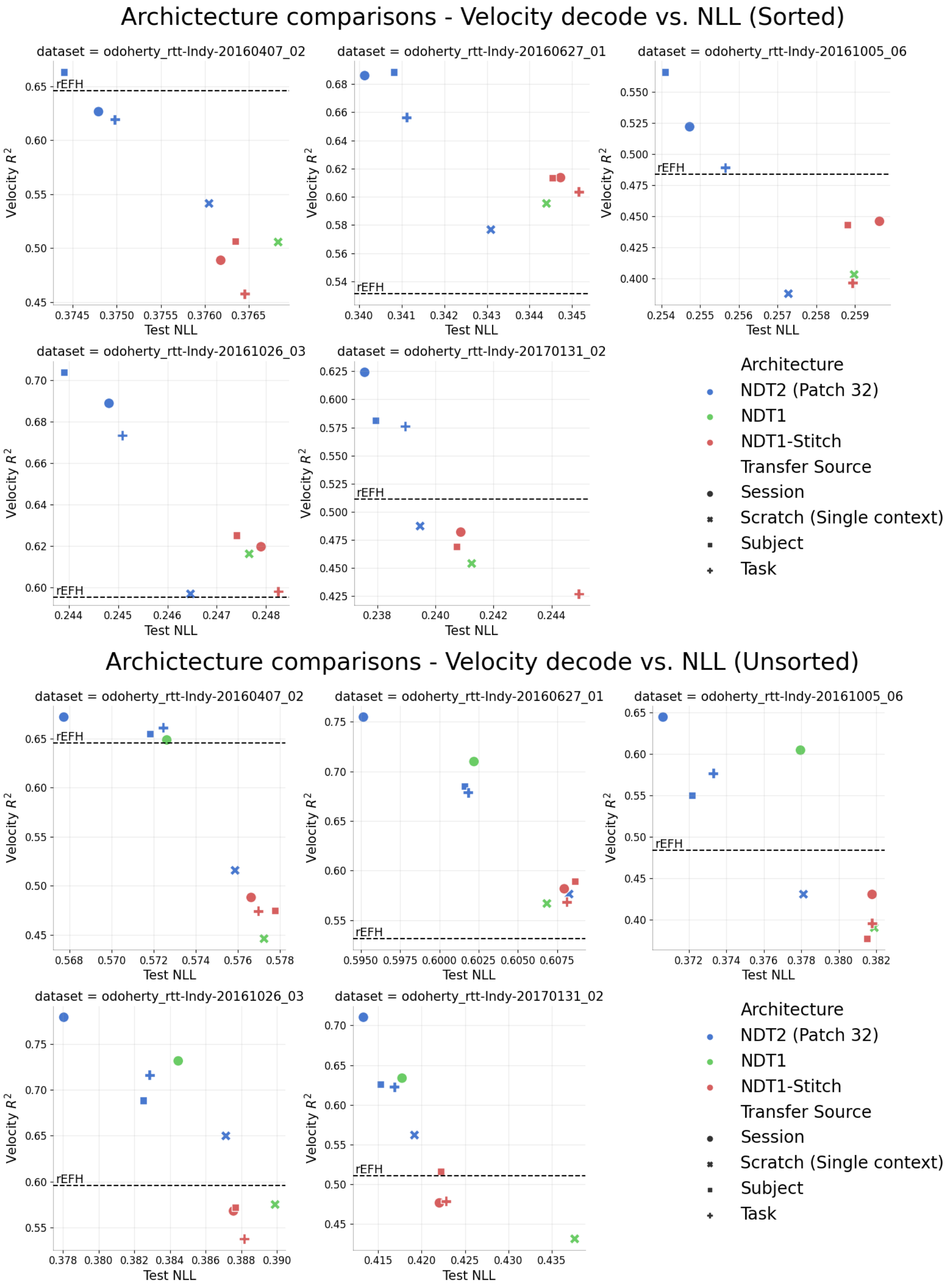

Figure 9. We breakout Fig. 2 into individual datasets (points indicate means on 3 seeds). NDT2 shows consistent improvements over stitching, single session baselines, and rEFH (in most cases), but the ranking between data sources shifts, particularly for decoding scores.

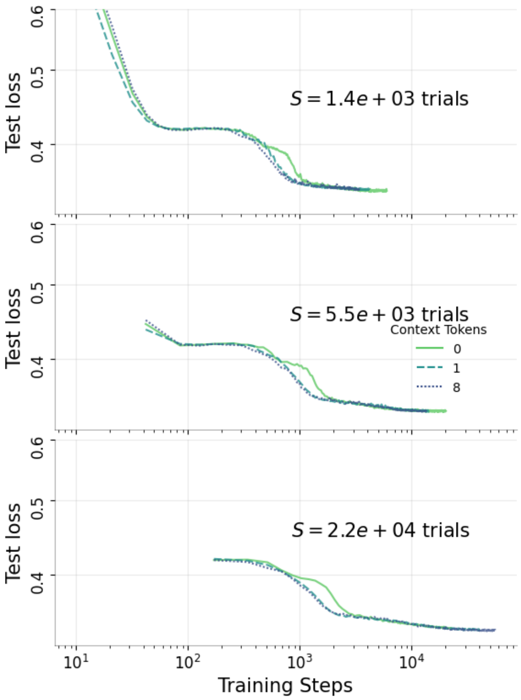

Figure 10. Session context tokens improve learning, as measured by unsupervised loss in model training curves, though the majority of benefit of realized with 1 token. We compare this for three data scales, annotated by S, where we scale the number of trials available per session. (The increments were 100%, 25%, and 6.25% of the data). We hypothesized that more data per session would make session tokens less relevant, but the primary effect of increasing convergence appears unchanged at these scales.

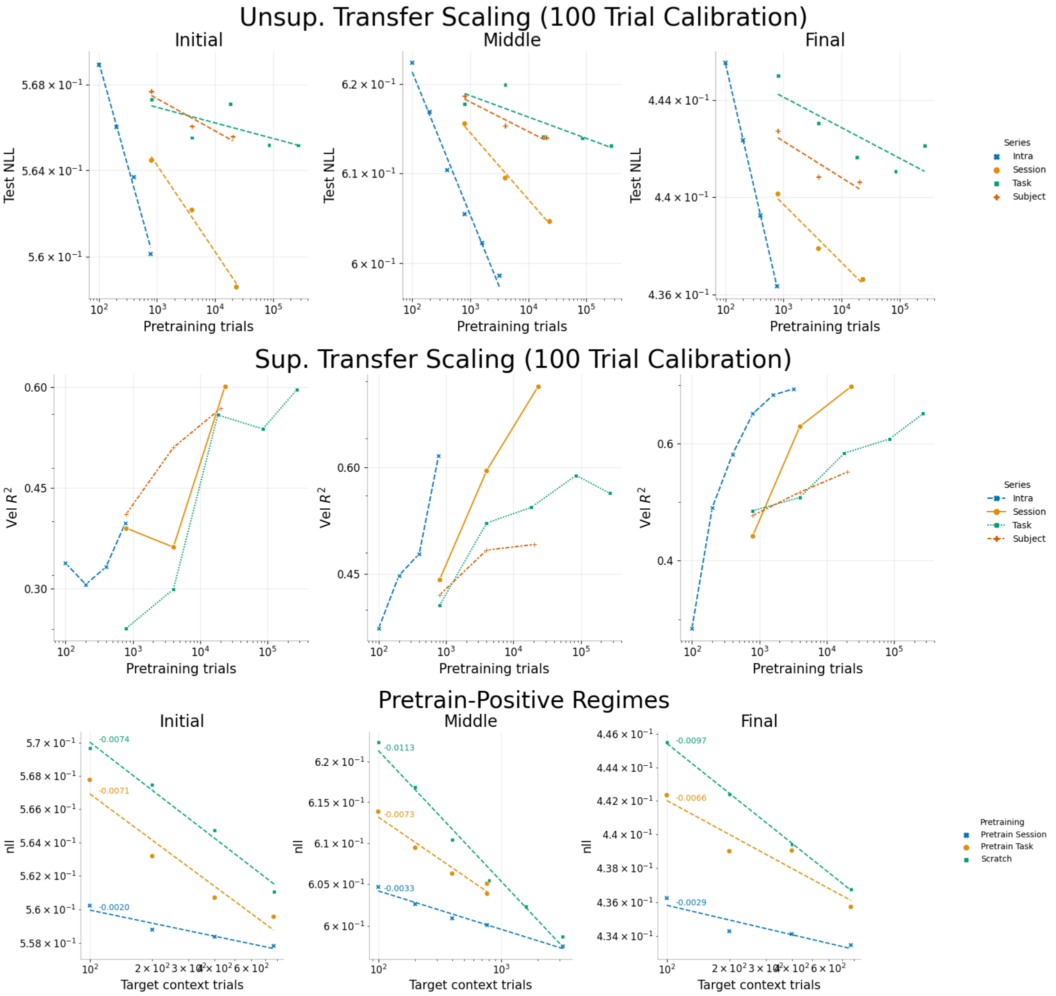

Figure 11. Individual dataset results for three scaling analyses. We presented the largest middle session in the primary text, here we also show results on the first and final sessions in the dataset. The unsupervised and supervised transfer scaling reiterate the previous conclusions: cross-subject and task transfer provides low returns on scaling for unsupervised reconstruction, and decoding results are much more optimistic than unsupervised results. For convergence analysis (Pretrain-positive regimes), all three trend lines suggest convergence beyond 1K trials.

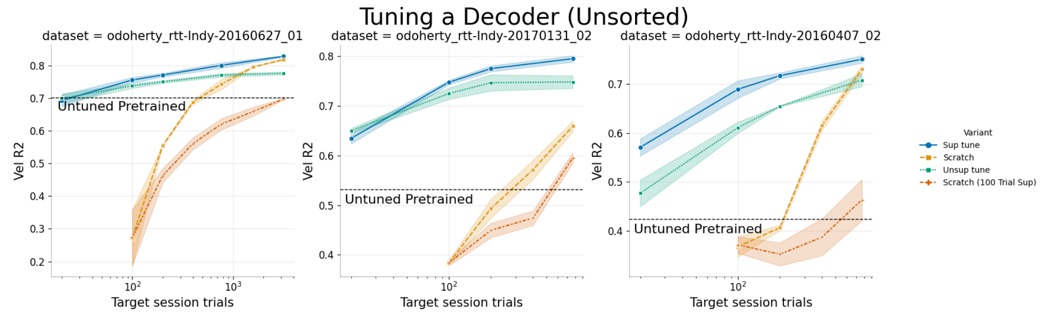

Figure 12. The same breakout as above for pretrained decoder tuning. Again, we find that decoder tuning reliably outperforms non-adaptive decoders.