LEARNING PREDICTIVE REPRESENTATIONS OF NEURAL POPULATION DYNAMICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural interfaces and population analyses increasingly require representations that can support several uses without learning a new neural encoder for each task. A central challenge is to learn such states from neural activity itself while retaining information about how the population evolves. We introduce Neural Predictive Coding (NPC), a self-supervised framework that uses future population activity as the learning signal. A causal encoder summarizes past and present activity into a compact state and contrastively predicts learned future targets across multiple timescales. Across motor control, spatial navigation, visual perception, and cognitive timing in three species, frozen NPC states improve forecasts of neurons excluded from pretraining when added to recent-activity summaries. These states also permit task decoding across all four functions, including mean R^2 of 0.50 for motor velocity and 0.58 for spatial position. Task-associated trajectories admit fitted updates that improve neural forecasts over static states in every function; in vision, evolved NPC states yield a 60% higher mean likelihood-gain score than CEBRA-Time. In a separate source-disjoint mouse cohort, pretraining improves likelihood-gain scores by 7% over scratch training at matched target updates and full query coverage, although longer scratch training performs better. Future-neural prediction thus provides a common training task for reusable readouts and quantitative analysis of population-state evolution.

1 INTRODUCTION

Neural population recordings inform brain computation and enable brain–computer interfaces that restore communication (Willett et al., 2023). Most decoders target a particular output, such as movement or speech. A reusable neural state could serve several readouts after common pretraining. Population structure persisting across behaviors and days motivates this prospect (Gallego et al., 2018; 2020). Advances in low-dimensional analysis, dynamical modeling, and neural pretraining bring this goal closer (Cunningham & Yu, 2014; Vyas et al., 2020; Ye et al., 2023). What should a representation retain before its downstream task is specified?

Neural activity is rich and continuous, whereas behavioral annotations are selective and costly. Supervised training therefore emphasizes information relevant to the labeled task. Self-supervised learning instead uses neural activity to define the training task before fitting downstream readouts (Azabou et al., 2021; Schneider et al., 2023; Ye et al., 2023). This separation also underlies masked and contrastive learning in vision (He et al., 2022; Chen et al., 2020). An online state must use past and present inputs, but future observations can guide what it learns to retain.

Current approaches make different choices about what to preserve. Masked and reconstruction objectives recover neural observations, enabling denoising, dynamical inference, and pretraining (Pandarinath et al., 2018; Ye et al., 2023). Contrastive objectives preserve specified temporal or behavioral associations (Schneider et al., 2023). Autoregressive models predict subsequent neural observations (Antoniades et al., 2024), while predictive representation methods target information shared by past and future (van den Oord et al., 2018; Clark et al., 2019). Reconstruction does not necessarily prioritize future-relevant features, and contrastive learning depends on the association chosen for training. We ask whether contrasting neural futures across horizons yields a compact online state useful for forecasting and downstream tasks.

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

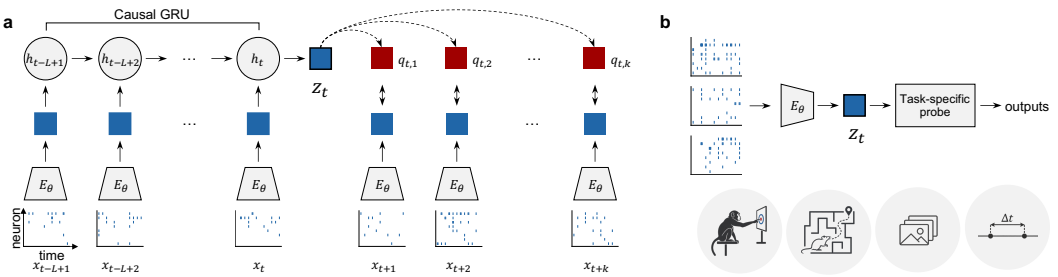


Figure 1: Learning neural states from future activity. (a) A neural encoder and causal GRU summarize activity up to the present; horizon-specific predictors contrast the corresponding future targets with sampled alternatives. (b) Frozen states support task readouts. Future activity supplies training targets only, and behavioral labels do not train the encoder.

We introduce Neural Predictive Coding (NPC), a neural-population framework built on contrastive predictive coding (van den Oord et al., 2018). A causal encoder summarizes recent activity; horizon-specific predictors distinguish future neural targets from alternatives sampled using population counts and sequence structure (Figure 1). We evaluate frozen states through neural forecasts, task readouts, and fitted evolution across four brain functions. A separate source-pretraining study tests adaptation to unseen visual mice. Our contributions are:

1. We propose a self-supervised neural representation framework that learns compact causal states by predicting future population activity across multiple timescales. Architecture-matched controls test the role of the predictive association.
2. We evaluate frozen representations across motor control, spatial navigation, visual perception, and cognitive timing in rats, mice, and macaques, using neural forecasting and task decoding across 26 recordings, with a separate test of adaptation to unseen mice.
3. We resolve task-associated neural trajectories in the learned state space and quantify their predictive evolution through fixed-emission forecasts, comparing fitted state updates with static states, CEBRA-Time, and direct neural history.

2 Related Work

Neural representation learning. Self-supervised neural representations differ in the information their learning task asks them to retain. Gaussian-process factor analysis extracts smooth single-trial population trajectories (Yu et al., 2009). NDT models population activity without recurrent dynamics (Ye & Pandarinath, 2021); NDT2 extends neural pretraining across recording contexts (Ye et al., 2023); CEBRA uses behavioral and/or temporal associations to organize embeddings (Schneider et al., 2023). Dynamical Components Analysis (DCA) learns linear projections by maximizing past–future predictive information (Clark et al., 2019), and CPC contrasts predicted future latents with alternatives (van den Oord et al., 2018). NPC adds nonlinear causal neural contexts and population-conditional contrasts. Our comparisons include causal CEBRA-Time, a local causal NDT2 adapter, and objective controls, but not DCA. Our challenge is to connect this training criterion to future-neural and task utility under online input access.

Neural decoding and population dynamics. Neural prediction, decoding, and dynamical analyses reveal complementary population structure. Reaching trajectories motivate dynamical descriptions (Churchland et al., 2012), while single-trial dynamics and latent-space alignment can improve neural interfaces (Kao et al., 2015; Degenhart et al., 2020). POYO scales supervised population decoding across recordings (Azabou et al., 2023), while stimulus-conditioned foundation models predict responses to new stimulus types (Wang et al., 2025). LFADS infers latent dynamics through a generative reconstruction model (Pandarinath et al., 2018); behavior-aware dynamical modeling separates behaviorally relevant and other dynamics (Sani et al., 2021; 2024), and MARBLE characterizes local dynamical structure geometrically (Gosztolai et al., 2025). The Neural Latents Benchmark provides shared held-out neural prediction tasks (Pei et al., 2021). These studies differ in supervision,

input access, and transfer setting. Our challenge is to relate learned state organization to predictive evolution through independent neural endpoints and history/task references.

3 METHODS

NPC learns an online neural state by predicting future population activity. Its encoder observes only past and present inputs.

3.1 NEURAL AND TEMPORAL ENCODING

Let $x_t \in \mathbb{N}^N$ contain input-neuron counts in bin t. A per-bin neural encoder E_θ extracts population features, and a causal recurrent encoder g_ψ integrates their recent history:

$$z_t = \mathrm{norm}\big[P\, g_\psi\big(E_\theta(x_{t-L+1}), \ldots, E_\theta(x_t)\big) + b_P\big]\,, \qquad u_{t+h} = \mathrm{norm}[E_\theta(x_{t+h})]\,. \tag{1}$$

Here E_θ includes training-fitted $\log(1+x)$ standardization, g_ψ returns the final GRU output, (P, b_P) projects it into the state space, and norm denotes unit normalization. The future target shares E_θ with the context branch and receives gradients. A horizon-specific affine predictor gives $q_{t,h} = \mathrm{norm}(W_h z_t + b_h)$.

3.2 MULTIHORIZON PREDICTIVE OBJECTIVE

For query t and horizon h, the candidate set $\mathcal{C}_{t,h}$ contains its true future and sampled alternatives. With $K_{t,h}$ candidates including the positive, NPC minimizes

$$\mathcal{L}_{\mathrm{NPC}} = -\frac{1}{|\mathcal{H}|} \sum_{h \in \mathcal{H}} \mathbb{E}_t \left[\frac{1}{\log K_{t,h}} \log \frac{\exp(q_{t,h}^{\mathsf{T}} u_{t+h}/\tau)}{\sum_{j \in \mathcal{C}_{t,h}} \exp(q_{t,h}^{\mathsf{T}} u_j/\tau)} \right]. \tag{2}$$

Dividing by $\log K_{t,h}$ normalizes losses with different numbers of candidates. To adapt the contrasts to neural activity, the sampler preferentially combines temporally separated alternatives within a sequence with population-count-matched alternatives across sequences. The count-matched alternatives are intended to discourage solutions based only on overall population rate. An expanded candidate bank supplies bounded negatives, with fallback sampling for insufficient preferred pools. These preferences define the learning task rather than guaranteeing removal of rate or temporal cues. Because negatives are conditionally sampled, we do not interpret the objective as a mutual-information bound.

3.3 NEURAL PRETRAINING

Only input-neuron activity enters representation training; behavioral labels and neurons reserved for evaluation are excluded. Training-fitted preprocessing and sequence-safe contexts prevent future bins or sequence gaps from entering an online state. The main implementation learns a 128-dimensional state from 16-dimensional per-bin features and trains at 100-, 200-, and 400-ms horizons. AdamW optimizes the neural encoder, GRU, projection, and prediction heads jointly. Checkpoints and learning rates within each configuration use sequence-isolated inner calibration. Appendix B specifies population sampling, the motor encoder variant, and all optimization settings.

3.4 DOWNSTREAM READOUTS AND FITTED EVOLUTION

For within-recording evaluation, neural forecasting, task readout, and fitted evolution each use the frozen pretrained state. Task labels fit the readout only. For dynamics, a training-fitted 16-dimensional projection $v_t = T(z_t)$ supports a separately fitted affine transition G_h at each horizon. We fit a concurrent Poisson emission map e from current states to reserved-neuron counts, then compare

$$\widehat{\lambda}^{\mathrm{static}}_{t+h} = e(v_t), \qquad \widehat{\lambda}^{\mathrm{evolved}}_{t+h} = e(G_h(v_t))\,. \tag{3}$$

The shared map and future targets isolate the state update's readout value. These horizon-specific maps make direct forecasts; they are not autonomous rollouts.

Table 1: Main benchmark and task readouts. Three seeds per method and recording; motor sessions are averaged within animal.

Function	Species	Task readout	Animals	Recordings
Motor control	Macaque	Velocity	4	12
Navigation	Rat	Position	4	4
Visual perception	Mouse	Image identity	6	6
Cognitive timing	Macaque	Produced interval	4	4

3.5 Adapting to a new recording

A separate visual transfer experiment learns a shared source encoder and adapts its input interface to each target recording, while freezing the pretrained core. The source and target mice are disjoint. Only neural activity trains the interface; reserved neurons are used later to fit the forecasting readout. The sampling fraction restricts positive query anchors; negative candidates and rate-stratification statistics still use the full target-training recording. Downstream readouts also use a fixed target-training set. This design tests representation adaptation under restricted query sampling, not restricted access to target neural data.

4 Experiments

Datasets. We evaluate four brain functions in rats, mice, and macaques (Table 1). The main benchmark contains 26 recordings, with a separately trained encoder for each recording. Motor data use the processed Perich archive (Perich et al., 2025); navigation uses hippocampal recordings (Grosmark & Buzsáki, 2016; Grosmark et al., 2016); vision uses Allen Visual Behavior Neuropixels (Bennett et al., 2026); timing includes Ready–Set–Go and Cue–Set–Go (Sohn et al., 2019; Wang et al., 2020). Figures 2–4 share 234 recording–method–seed checkpoints. Random-initialization, objective, and adaptation studies have separate scopes (Appendix A).

Tasks and metrics. Regularized Poisson readouts predict future counts of neurons excluded from representation training. The main-benchmark neural score is likelihood gain over training-derived constant rates, in bits/spike (Pei et al., 2021). Incremental prediction compares recent-activity features with the same features augmented by the frozen state. These features contain current counts and trailing 100-ms, 200-ms, and full-context means after $\log(1 + x)$. The comparison tests accessibility to this readout, rather than information absent from the complete history. Task probes recover the variables in Table 1, using a common inner-selected probe menu and nested whole-sequence label budgets. Selection labels count toward each budget. For dynamics, we pair trajectory views with fixed-emission forecasts and history/task references.

Implementation. Models receive 300-ms causal histories in 20-ms bins, except navigation, which uses 500-ms histories and 25-ms bins. NPC and causal CEBRA-Time use 128-dimensional states; NDT2 uses its official 256-dimensional backbone with a local causal adapter, trained from random initialization. These comparisons assess the stated implementations, not published pretrained systems. A separate diagnostic compares six objectives on five recordings with three paired seeds and common initialized trunks, neurons, queries, and 4,000 updates (Appendix E).

Cross-recording adaptation. The separate visual study pretrains on five sessions from three mice and evaluates sixteen sessions from eight disjoint validation mice. It compares 250 interface-adaptation updates after 5,000 source updates with target-only representation training for 250 or 5,250 updates, at six positive-query fractions from 1 to 100%. All arms retain the full target-training negative bank and rate strata, and all readouts use 8,192 target-training anchors. Scores average 20–320-ms horizons and use validation-neuron mean rates as the constant-rate reference; this historical endpoint differs from the main benchmark (Appendix G).

Statistics and evaluation scope. We average seeds within recording and sessions within animal; 95% animal-bootstrap intervals describe biological variation. Individual-recording analyses have

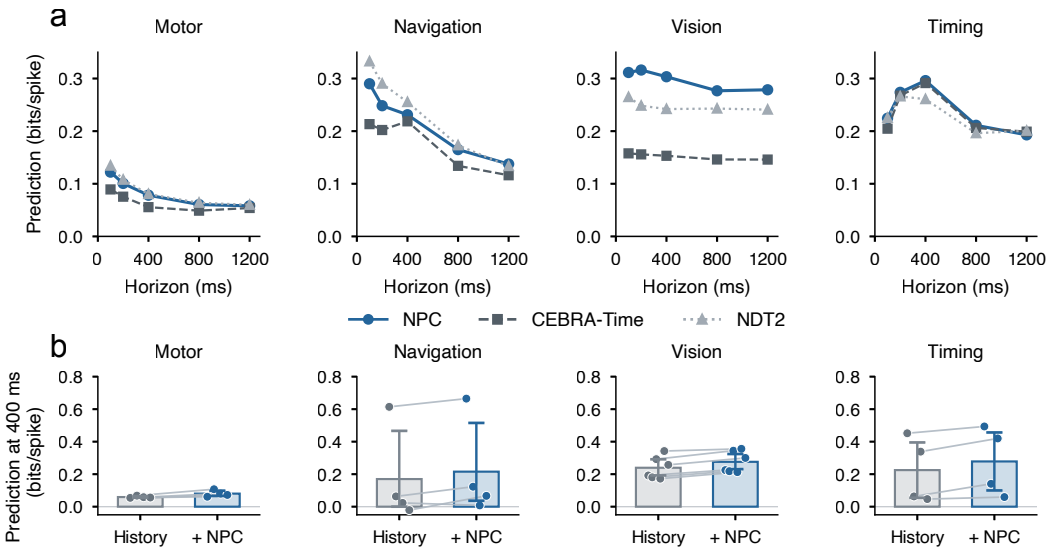


Figure 2: Future-neural prediction from frozen states. (a) Excluded-neuron likelihood above a constant-rate reference across prediction horizons. (b) Recent-activity readouts with and without NPC at 400 ms. Curves/bars show animal means; paired points are animals, and whiskers are 95% animal-bootstrap intervals. Methods share targets and query times; Table 1 gives the cohort.

no population interval. Matched comparisons preserve targets and query times; unavailable label-budget estimates are not imputed. Exact paired animal sign-flip tests use Holm correction over four incremental neural comparisons and, separately, eight full-label task and eight evolved-neural contrasts. NPC motor encoding and navigation checkpoint selection received additional tuning after result inspection. These within-recording comparisons are therefore exploratory development evaluations, with unequal tuning, dimensions, and compute. Appendix C specifies the splits, probes, and uncertainty calculations.

5 Results

5.1 Predictive states forecast activity beyond the training neurons

We first tested the defining requirement of a predictive state: useful forecasts of neural activity beyond the neurons that trained it. NPC states predicted this excluded-neuron activity across all four functions (Figure 2a). At 400 ms, mean NPC scores were positive in every function, with the largest score in vision (0.303 bits/spike over a constant-rate forecast). NPC exceeded CEBRA-Time in all four function means; compared with NDT2, it scored higher in vision and timing and lower in motor and navigation. Thus, forecasting performance depended on the function rather than following a uniform method ranking.

Adding NPC to recent-activity summaries improved mean 400-ms prediction in all four functions, from 0.021 bits/spike in motor to 0.053 in timing (Figure 2b). The small biological sample sizes and adaptive development limit inference: none of the four exact paired animal sign-flip tests reached 0.05 after Holm correction.

5.2 Frozen states decode task variables across four brain functions

Frozen NPC states recover motor velocity, spatial position, visual image identity, and produced timing intervals through task-specific readouts. Example predictions show what these readouts recover (Figure 3a), while animal-level scores quantify their performance (Figure 3b). At full labels, NPC reached mean $R^2 = 0.496$ for motor, 0.578 for navigation, and 0.224 for timing, with visual balanced accuracy of 0.624. NPC exceeded CEBRA-Time in each function mean. NDT2 scored higher in

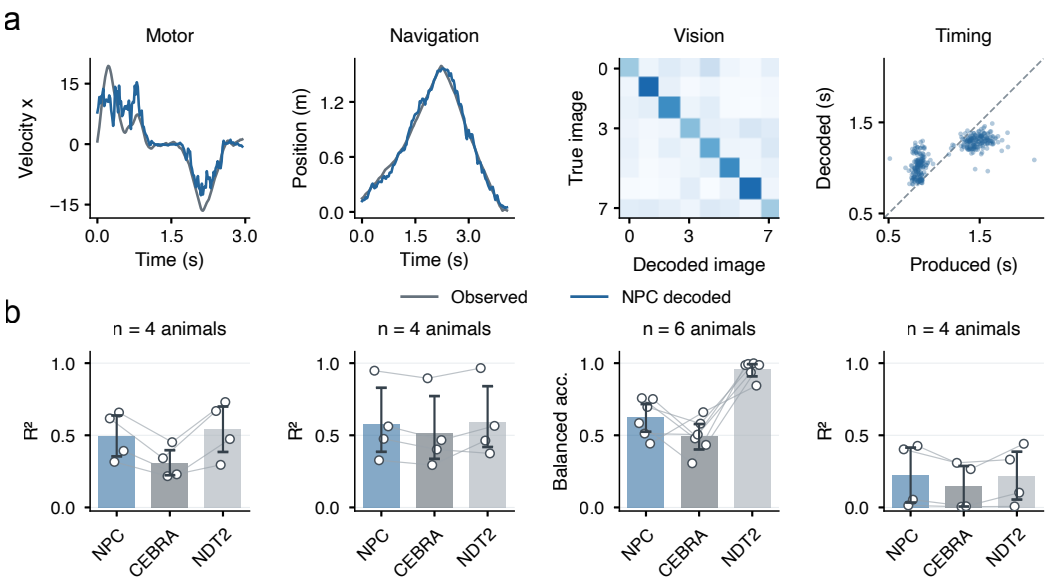


Figure 3: Task decoding from frozen neural states. (a) Fixed decoding examples; the visual confusion matrix is row-normalized (0–1). (b) Full-label R^2 for regression tasks and balanced accuracy for image identity. Bars show animal means, connected points pair animals across methods, and whiskers show 95% animal-bootstrap intervals. Encoders receive no task labels.

motor, navigation, and vision, with a small reverse difference in timing. None of the eight exploratory paired-animal comparisons reached 0.05 after Holm correction (Appendix C).

The label-budget and session analyses test how readily these states can be used with different amounts of supervision and across recordings. Only timing supports the 1% sequence-label budget; the 5% budget additionally supports two motor animals. Unavailable estimates reflect sequence-calibration or class-coverage requirements, not a different set of requested fractions. All 12 motor sessions have positive NPC R^2 (0.212–0.790), with a separately trained encoder and probe in each session. Individual-animal curves and session results are reported in Appendix C.

Neural prediction and task readout offer complementary views of the state. In the same six visual mice, NPC had a higher 400-ms future-neural score than NDT2, whereas NDT2 made image identity more accessible. A separate seven-recording linear-probe screen sharpens this distinction: trained NPC improves motor, navigation, and timing readout relative to matched random encoders, but reduces image accuracy in both visual recordings despite improving their neural forecasts (Appendix D). Predictive training therefore need not maximize accessibility of every task variable.

5.3 Learned trajectories admit predictive state evolution

We next asked whether the organization of NPC states permits useful temporal updates. Training-fitted projections show task-associated trajectories without behavioral supervision (Figure 4a). The quantitative analysis uses the same 26 recordings and 234 checkpoints as the neural and task benchmarks, including all four timing animals.

We compare evolved states through a common neural endpoint, avoiding rankings based on distances in different latent spaces. At 400 ms, evolved NPC scored higher than CEBRA-Time in navigation, vision, and timing, and lower in motor (Figure 4b). The largest difference was in vision: 0.259 versus 0.162 bits/spike, a 60% higher mean score. Five of six mice favored NPC, although the paired test did not reach the Holm-adjusted 0.05 threshold ($p = 0.50$ across eight contrasts). Direct-history prediction remained stronger in motor and navigation; NPC had higher mean scores in vision and timing (Appendix F, Figure 15).

Fitted evolution improved NPC's static forecast in every function mean (Figure 4c). Mean gains were 0.191, 0.284, 0.434, and 0.078 bits/spike for motor, navigation, vision, and timing, respectively. The

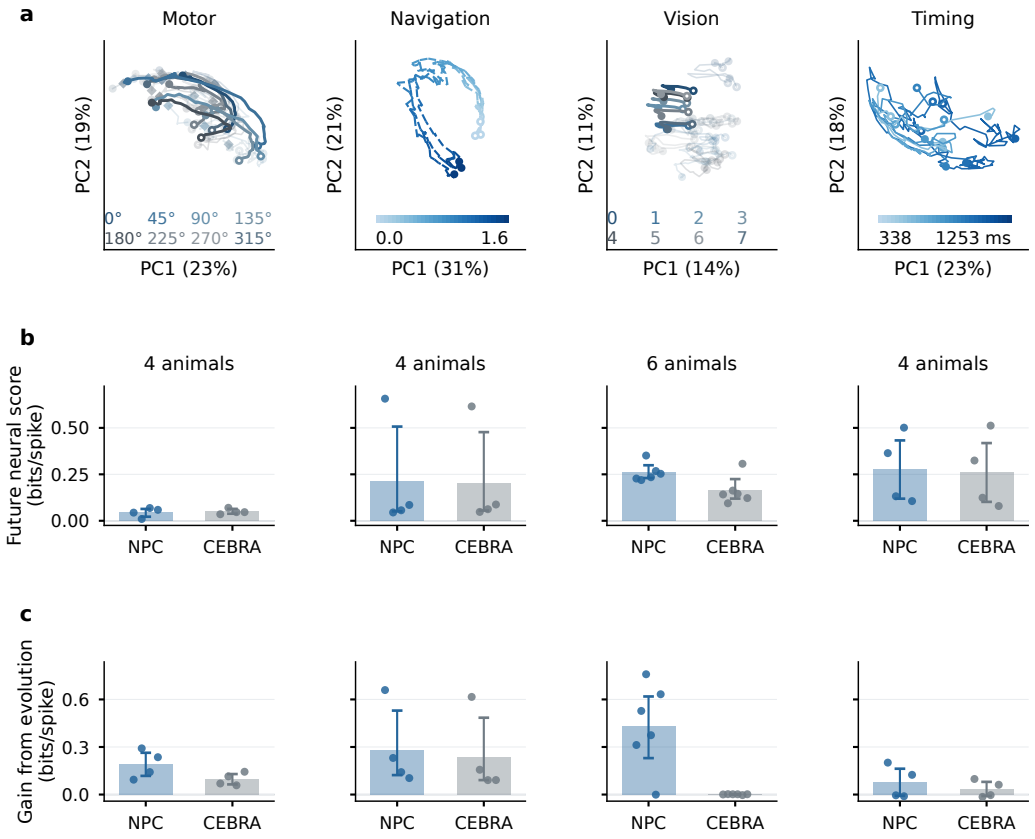


Figure 4: Neural-state organization and predictive evolution. (a) Task-colored NPC trajectories; filled/open circles mark starts/ends. (b) Evolved neural scores for NPC and CEBRA-Time. (c) Improvement over each method's static forecast. Bars show animal means, points individual animals, and whiskers 95% animal-bootstrap intervals at 400 ms.

visual gain partly reflects recovery from a weak static forecast: NPC increased from -0.175 to 0.259, whereas CEBRA-Time changed from 0.161 to 0.162. These comparisons establish the readout value of a fitted update, rather than autonomous neural dynamics. Task progression and overlapping context windows remain alternative explanations for predictable geometry (Appendix F).

5.4 Pretraining improves adaptation at matched target updates

We tested neural forecasting in mice absent from source training while varying the positive queries used to adapt each target interface. Every arm still accesses the full target-training recording for negative candidates and rate strata, and all readouts use the same 8,192 training anchors. At the 1% query fraction, pretrained NPC scored 0.225 bits/spike versus 0.200 for target-only training with the same 250 target updates (Figure 5). The ratio of animal-mean scores was 1.12 (95% paired mouse-bootstrap interval 1.08–1.19). All eight mice improved; the paired gain was 0.025 bits/spike [0.015, 0.036], with Holm-adjusted $p = 0.0469$ over six fractions. The longer 5,250-update target-only control scored 0.151, giving a larger 49% relative advantage for pretraining.

The comparison changes as positive-query coverage increases. At full query coverage, pretraining remained 7% above the 250-update control, with a paired gain of 0.022 bits/spike [0.011, 0.034] and improvements in all eight mice. Figure 5b shows this comparison separately for each unseen mouse, averaging all seeds and sessions. Against longer target-only training, the mean advantage shrank to 0.010 bits/spike at 5% and reversed from 10% onward. At 10%, all eight mice favored that

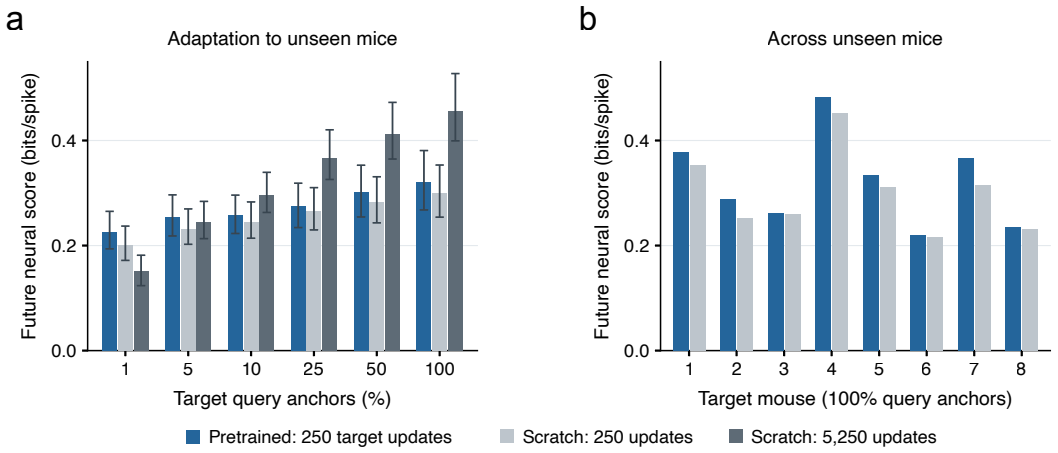


Figure 5: Neural forecasting after adaptation to unseen visual mice. (a) Pretrained interface adaptation and target-only training across six positive-query fractions. (b) Full-query scores for each unseen mouse, comparing pretraining with 250-update scratch training; bars average three seeds and two sessions. Whiskers in (a) show 95% mouse-bootstrap intervals across eight mice. Fractions restrict queries only; all arms use the full target-training negative bank and fixed readout data.

Table 2: Common-trunk objective controls. Neural gains are NPC minus control, averaged across five recordings after seeds and horizons; win counts use recording means. All comparisons match updates, not compute.

Control	Neural gain (bits/spike)	Neural wins (of 5)	Task wins (of 5)
Temporal contrast	$+0.078$	5	5
Future-count prediction	$+0.079$	5	5
Shuffled future	$+0.168$	5	5
Single horizon	$+0.023$	5	4
Global negatives	$+0.003$	2	4

control, with a mean pretrained-minus-control difference of -0.038 bits/spike $[-0.053, -0.023]$. Thus, source pretraining helps this restricted-query regime, but the experiment does not establish reduced target-data requirements. The smallest-query emphasis is a post-hoc development finding on source-disjoint validation mice; the original 10%/25% transfer gate was not met (Appendix G).

5.5 Controls identify what predictive training contributes

Learning the correct association between context and future improves both neural prediction and task readout. In the common-trunk experiment, NPC scored higher than shuffled-future training in all five recordings and all three paired seeds (Table 2). The neural advantage also held separately at all five native horizons. NPC similarly exceeded the temporal-contrast and direct future-count controls on both endpoints in every recording. A separate random-encoder screen found that trained NPC added more neural predictive value beyond identical history features in all seven recordings (Appendix D). These comparisons link the observed utility to learning, beyond simply adding a feature vector.

The design choices have different effects. Multiple horizons improved neural prediction in all five recordings and task scores in four. Population-aware sampling had mixed effects relative to global negatives: it improved both endpoints in motor and navigation, improved visual task accuracy while reducing visual neural prediction, and reduced both timing outcomes. The latter comparison is the closest common-trunk control for the underlying CPC construction. Thus, the true future association and multiple timescales are more consistently useful in this diagnostic than the structured negative sampler.

The controls match initialized trunks, neurons, query samples, and 4,000 updates, but neither compute nor convergence. NPC took 20.7–29.4 times the recorded training time of count prediction. The temporal control also changes head tying and target offsets; it does not isolate time direction or reproduce official CEBRA-Time. These are five-recording diagnostics with technical seeds, not five replicated biological cohorts. Appendix E reports every recording and task endpoint.

6 Discussion

The predictive task defines what an online state should retain before a behavioral readout is chosen. By contrasting learned futures at several horizons, NPC trains a compact context to distinguish subsequent population states. This supplies a temporal criterion for representation learning while leaving the downstream variable unspecified. The common-trunk comparisons suggest that the useful signal lies in the correspondence between context and future, with less consistent gains from changing the negative distribution. This distinction motivates treating the predictive association as the central design choice and evaluating sampling strategies separately.

The evaluation separates three consequences of this choice. Future-neural readouts test information accessible about neurons excluded from pretraining; task probes test accessibility of behaviorally relevant variables; fixed-emission forecasts test whether updating the state improves prediction while keeping the neural readout unchanged. Their combination connects representation learning to population dynamics beyond the appearance of trajectories. Reuse here means fitting these readouts to the same frozen encoder within a recording. Across functions, the shared element is the learning task and evaluation procedure. This distinction offers a practical development path: learn neural states before selecting downstream readouts, then assess cross-recording transfer as a separate capability.

Predictive states need not maximize task accessibility. NDT2 makes visual image identity more accessible despite NPC's stronger future-neural forecasts. The trained-versus-random visual controls show the same tension. A representation's value therefore depends on its intended readout, not a single cross-task ranking.

Pretraining helps adaptation to new mice, most strongly at the smallest query fraction; longer target-only training wins as query coverage increases. Full-recording negative candidates and rate strata, together with fixed readout calibration, make this an adaptation study rather than a data-efficiency test. Whether pretraining also transfers a fitted state-transition model remains an open question.

The cohorts are small and the evaluations remain exploratory: NPC received additional tuning, model dimensions differ, and objective controls match updates rather than compute or convergence. Biological population structure can constrain learning (Sadtler et al., 2014), but fitted state transitions alone do not identify that structure; history and task progression can explain substantial predictive geometry. Independent cohorts and compute-matched controls are needed to test how broadly the observed benefits extend.

7 Conclusion

NPC learns compact online states from future neural population activity without behavioral supervision. Across four brain functions, frozen states support forecasting and decoding, while fixed-emission tests quantify the value of fitted state evolution. Source pretraining also improves prediction in unseen visual mice at matched target-training updates, although longer training from scratch performs better. These uses share a predictive training principle, but require distinct tests of task accessibility, temporal utility, and transfer.

AI USE STATEMENT

AI assistants supported literature search, manuscript drafting and revision. They also assisted in framing the research question and interpreting completed analyses. The authors are responsible for the scientific claims, references, and final submission.

REPRODUCIBILITY STATEMENT

The appendix specifies the representation objectives, temporal-access constraints, evaluation splits, readout selection, aggregation units, and scope of each study. The accompanying source package records figure sources and numerical claims. Separate development analyses are identified explicitly.

REFERENCES

Allen Institute for Brain Science. Visual behavior neuropixels: AllenSDK documentation. Dataset documentation, n.d. URL https://allensdk.readthedocs.io/en/latest/visual_behavior_neuropixels.html. Accessed September 26, 2026.

Antonis Antoniades, Yiyi Yu, Joseph Canzano, William Wang, and Spencer LaVere Smith. Neuroformer: Multimodal and multitask generative pretraining for brain data. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2311.00136.

Mehdi Azabou, Mohammad Gheshlaghi Azar, Ran Liu, Chi-Heng Lin, Erik C. Johnson, Kiran Bhaskaran-Nair, Max Dabagia, Bernardo Avila-Pires, Lindsey Kitchell, Keith B. Hengen, William Gray-Roncal, Michal Valko, and Eva L. Dyer. Mine Your Own vieW: Self-supervised learning through across-sample prediction. *arXiv preprint arXiv:2102.10106*, 2021. URL https://arxiv.org/abs/2102.10106.

Mehdi Azabou, Vinam Arora, Venkataramana Ganesh, Ximeng Mao, Santosh Nachimuthu, Michael J. Mendelson, Blake Richards, Matthew G. Perich, Guillaume Lajoie, and Eva L. Dyer. A unified, scalable framework for neural population decoding. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://openreview.net/pdf?id=sw2Y0sirtM.

Corbett Bennett, Samuel D. Gale, Greggory Heller, Tamina K. Ramirez, Hannah Belski, Alex Piet, Omid Zobeiri, Adam Amster, Anton Arkhipov, Alex Cahoon, Shiella Caldejon, Mikayla Carlson, Linzy Casal, Scott F. Daniel, Colin Farrell, Marina Garrett, Ryan Gillis, Conor Grasso, Ben J. Hardcastle, Ross Hytnen, Tye Johnson, Peter Ledochowitsch, Quinn L'Heureux, Dana Mastrovito, Ethan G. McBride, Stefan Mihalas, Chris Mochizuki, Christopher B. Morrison, Chelsea Nayan, Nhan-Kiet Ngo, Kat North, Douglas R. Ollerenshaw, Ben Ouellette, Paul Rhoads, Kara Ronellenfitch, Martin Schroedter, Joshua H. Siegle, Cliff Slaughterbeck, David Sullivan, Jackie Swapp, Michael Taormina, Wayne Wakeman, Xana Waughman, Allison Williford, John W. Phillips, Peter A. Groblewski, Séverine Durand, Christof Koch, and Shawn R. Olsen. Map of spiking activity underlying change detection in the mouse visual system. *Cell*, 189:4775–4791.e13, 2026. doi: 10.1016/j.cell.2026.06.025.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607, 2020. URL https://proceedings.mlr.press/v119/chen20j.html.

Mark M. Churchland, John P. Cunningham, Matthew T. Kaufman, Justin D. Foster, Paul Nuyujukian, Stephen I. Ryu, and Krishna V. Shenoy. Neural population dynamics during reaching. *Nature*, 487: 51–56, 2012. doi: 10.1038/nature11129. URL https://www.nature.com/articles/nature11129.

David Clark, Jesse Livezey, and Kristofer Bouchard. Unsupervised discovery of temporal structure in noisy data with dynamical components analysis. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://papers.nips.cc/paper_files/paper/2019/hash/704cddc91e28d1a5517518b2f12bc321-Abstract.html.

John P Cunningham and Byron M Yu. Dimensionality reduction for large-scale neural recordings. *Nature Neuroscience*, 17:1500–1509, 2014. doi: 10.1038/nn.3776. URL https://www.nature.com/articles/nn.3776.

Alan D. Degenhart, William E. Bishop, Emily R. Oby, Elizabeth C. Tyler-Kabara, Steven M. Chase, Aaron P. Batista, and Byron M. Yu. Stabilization of a brain–computer interface via the alignment of low-dimensional spaces of neural activity. *Nature Biomedical Engineering*, 4:672–685, 2020. doi: 10.1038/s41551-020-0542-9. URL https://www.nature.com/articles/s41551-020-0542-9.

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*, 9:4233, 2018. doi: 10.1038/s41467-018-06560-z. URL https://www.nature.com/articles/s41467-018-06560-z.

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*, 23:260–270, 2020. doi: 10.1038/s41593-019-0555-4. URL https://www.nature.com/articles/s41593-019-0555-4.

Adam Gosztolai, Robert L. Peach, Alexis Arnaudon, Mauricio Barahona, and Pierre Vandergheynst. MARBLE: interpretable representations of neural population dynamics using geometric deep learning. *Nature Methods*, 22:612–620, 2025. doi: 10.1038/s41592-024-02582-2. URL https://www.nature.com/articles/s41592-024-02582-2.

A. D. Grosmark, J. Long, and G. Buzsáki. Recordings from hippocampal area CA1, PRE, during and POST novel spatial learning. CRCNS.org, hc-11 dataset, 2016. URL https://crcns.org/data-sets/hc/hc-11.

Andres D. Grosmark and György Buzsáki. Diversity in neural firing dynamics supports both rigid and learned hippocampal sequences. *Science*, 351:1440–1443, 2016. doi: 10.1126/science.aad1935.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16000–16009, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/He_Masked_Autoencoders_Are_Scalable_Vision_Learners_CVPR_2022_paper.html.

Jonathan C. Kao, Paul Nuyujukian, Stephen I. Ryu, Mark M. Churchland, John P. Cunningham, and Krishna V. Shenoy. Single-trial dynamics of motor cortex and their applications to brain-machine interfaces. *Nature Communications*, 6:7759, 2015. doi: 10.1038/ncomms8759. URL https://www.nature.com/articles/ncomms8759.

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*, 15:805–815, 2018. doi: 10.1038/s41592-018-0109-9. URL https://www.nature.com/articles/s41592-018-0109-9.

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. *arXiv preprint arXiv:2109.04463*, 2021. URL https://arxiv.org/abs/2109.04463.

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. DANDI Archive, dataset 000688, version 0.250122.1735, 2025. URL https://dandiarchive.org/dandiset/000688/0.250122.1735.

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: 423–426, 2014. doi: 10.1038/nature13665. URL https://www.nature.com/articles/nature13665.

Omid G. Sani, Hamidreza Abbaspourazad, Yan T. Wong, Bijan Pesaran, and Maryam M. Shanechi. Modeling behaviorally relevant neural dynamics enabled by preferential subspace identification. *Nature Neuroscience*, 24:140–149, 2021. doi: 10.1038/s41593-020-00733-0. URL https://www.nature.com/articles/s41593-020-00733-0.

Omid G. Sani, Bijan Pesaran, and Maryam M. Shanechi. Dissociative and prioritized modeling of behaviorally relevant neural dynamics using recurrent neural networks. *Nature Neuroscience*, 27: 2033–2045, 2024. doi: 10.1038/s41593-024-01731-2. URL https://www.nature.com/articles/s41593-024-01731-2.

Steffen Schneider, Jin Hwa Lee, and Mackenzie Weygandt Mathis. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 617:360–368, 2023. doi: 10.1038/s41586-023-06031-6. URL https://www.nature.com/articles/s41586-023-06031-6.

Hansem Sohn, Devika Narain, Nicolas Meirhaeghe, and Mehrdad Jazayeri. Bayesian computation through cortical latent dynamics. *Neuron*, 103(5):934–947.e5, 2019. doi: 10.1016/j.neuron.2019.06.012.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. URL https://arxiv.org/abs/1807.03748.

Saurabh Vyas, Matthew D. Golub, David Sussillo, and Krishna V. Shenoy. Computation through neural population dynamics. *Annual Review of Neuroscience*, 43:249–275, 2020. doi: 10.1146/annurev-neuro-092619-094115. URL https://doi.org/10.1146/annurev-neuro-092619-094115.

Eric Y. Wang, Paul G. Fahey, Zhuokun Ding, Stelios Papadopoulos, Kayla Ponder, Marissa A. Weis, Andersen Chang, Taliah Muhammad, Saumil Patel, Zhiwei Ding, Dat Tran, Jiakun Fu, Casey M. Schneider-Mizell, MICrONS Consortium, R. Clay Reid, Forrest Collman, Nuno Maçarico da Costa, Katrin Franke, Alexander S. Ecker, Jacob Reimer, Xaq Pitkow, Fabian H. Sinz, and Andreas S. Tolias. Foundation model of neural activity predicts response to new stimulus types. *Nature*, 640:470–477, 2025. doi: 10.1038/s41586-025-08829-y. URL https://www.nature.com/articles/s41586-025-08829-y.

Jing Wang, Eghbal Hosseini, Nicolas Meirhaeghe, Adam Akkad, and Mehrdad Jazayeri. Reinforcement regulates timing variability in thalamus. *eLife*, 9:e55872, 2020. doi: 10.7554/eLife.55872. URL https://elifesciences.org/articles/55872.

Francis R. Willett, Erin M. Kunz, Chaofei Fan, Donald T. Avansino, Guy H. Wilson, Eun Young Choi, Foram Kamdar, Matthew F. Glasser, Leigh R. Hochberg, Shaul Druckmann, Krishna V. Shenoy, and Jaimie M. Henderson. A high-performance speech neuroprosthesis. *Nature*, 620:1031–1036, 2023. doi: 10.1038/s41586-023-06377-x. URL https://www.nature.com/articles/s41586-023-06377-x.

Joel Ye and Chethan Pandarinath. Representation learning for neural population activity with Neural Data Transformers. *arXiv preprint arXiv:2108.01210*, 2021. URL https://arxiv.org/abs/2108.01210.

Joel Ye, Jennifer L. Collinger, Leila Wehbe, and Robert Gaunt. Neural data transformer 2: Multi-context pretraining for neural spiking activity. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/fe51de4e7baf52e743b679e3bdba7905-Paper-Conference.pdf.

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

A STUDY SCOPE AND DATA PROVENANCE

The paper combines distinct studies that address successive questions. Their checkpoints, cohorts, and evaluation endpoints are not interchangeable. Table 3 records these boundaries. "Causal" denotes access to bins ending at the present, after their counts have been observed; it is not a biological causal claim or a measurement of inference latency. Separate representations are trained for different recordings/functions. The cross-function comparison does not evaluate one checkpoint transferred across all functions.

Table 3: Evidence scope of the reported studies.

Study	Biological coverage	Interpretation
Figures 2–4	12 motor sessions/four animals; four rats; six mice; four timing animals	Same 234 recording/method/seed checkpoints; adaptively developed configurations; exploratory evaluation.
Random controls	Two motor sessions, two rats, two mice, one timing recording; seed zero	Historical checkpoints; matched random/trained task and 400-ms neural readouts, with capped neural anchors.
Objective controls	One motor recording, one rat, two visual mice, one timing recording; three seeds	Common-trunk diagnostic at a fixed update budget; no biological replication outside vision.
Figure 5	Five source sessions/three mice; sixteen target sessions/eight disjoint mice	864 arm/budget/session/seed cells; post-hoc validation analysis with fixed downstream readout data.

Data and biological units. Motor recordings come from the Perich collection in the processed motor archive used with POYO (Perich et al., 2025; Azabou et al., 2023); three sessions are selected per subject. The four navigation recordings use the rat hippocampal collection processed for CEBRA (Grosmark & Buzsáki, 2016; Grosmark et al., 2016; Schneider et al., 2023). Vision uses six development mice from the Allen Visual Behavior Neuropixels collection (Bennett et al., 2026; Allen Institute for Brain Science, n.d.). Timing includes two Ready–Set–Go animals (Haydn and Gershwin) and two Cue–Set–Go animals (A and D), using the author-released cortical recordings associated with Sohn et al. (2019) and Wang et al. (2020). The Cue–Set–Go analysis uses simultaneously recorded cortical populations, not a concatenation of independently recorded neurons or pooled thalamic/caudate populations. These paradigms differ: reproducing a measured interval and producing a cued target interval are reported separately as well as in the equal-animal aggregate.

Each study uses a fixed acquisition-sequence train/validation split. Motor retains the supplied archive assignments. Visual development recordings use an outcome-blind, session-keyed hash ordering of complete sequences, allocating 80% to training and the remainder to validation. The original Haydn timing split is retained; added timing animals use deterministic 80/20 trial splits within task condition. Gershwin includes rewarded standard trials, whereas the added Cue–Set–Go recordings include all completed trials, rewarded or unrewarded. These eligibility differences remain part of the timing comparison. Input and target neurons are partitioned before representation learning. Every retained context and future target must belong to an eligible sequence; time gaps break trajectories. The main forecast evaluation uses matched query anchors and target neurons for all compared methods and horizons. Normalization is fitted on the training partition associated with the relevant model/readout fit. Task labels are used in probe fitting, not in the neural training loss, although the wider development process has been informed by inspected results. Frozen independent visual-test decisions from earlier work are not recomputed or reinterpreted by these development comparisons.

Source agreement. Figures 2–4 share all 234 checkpoint identities from 26 recordings and 18 animals. The dynamics analysis includes 75 newly evaluated cells: 48 final NPC motor/navigation checkpoints and 27 cells from the three additional timing animals. The other 159 cells retain their matching sources. Checkpoint hashes, paired anchors, targets, and history features were verified for all 234 cells; no animals were excluded according to effect direction. Figure 5 uses an independently specified source-pretraining protocol.

B TRAINING DETAILS AND COMPARISON IMPLEMENTATIONS

NPC parameterization. NPC inputs use training-fitted standardization of $\log(1 + \text{count})$, with standard deviations floored at 0.05. The dense per-bin encoder has a 64-dimensional hidden layer and a 16-dimensional output. A GRU maps these features to a 128-dimensional context, followed by a 128-dimensional affine projection and unit normalization. The future single-bin targets share the per-bin encoder and are unit normalized; each horizon predictor is a separately learned affine map from the state to the 16-dimensional target space, followed by unit normalization. Target gradients are propagated; there is no momentum teacher or stop-gradient target in this implementation. The training loss contains no behavioral objective or held-out-neuron target loss.

Motor uses a rate-preserving encoder variant: a dense network without a final output normalization is augmented by an affine projection of $\log(1+\text{population count})$. The recurrent state and future targets are normalized at their respective outputs. This change jointly modifies normalization placement and introduces a population-rate skip; the experiment does not isolate these two mechanisms. The motor dense-reference comparison matches parameter count. Navigation retains dense encoding with a refined checkpoint schedule. Vision and timing retain their original encoding variants.

Candidate construction. The main benchmark trains at physical horizons 100, 200, and 400 ms with temperature $\tau = 0.2$ and up to 64 negatives per query. Query positives occupy the first part of the target bank. Additional eligible targets are sampled without replacing those query positives, allowing query-specific selection from a larger pool. Approximately half the preferred negatives are temporally separated targets within the query sequence; the remaining preferred targets come from other sequences with the same population-count decile. The exclusion parameter is 80 ms, with integer conversion to three 25-ms bins (75 ms) in navigation. These experiments do not use the generic implementation's optional data-estimated autocorrelation exclusion radius.

Candidates matching that query's positive target identity or anchor are excluded; this is not a global deduplication of all negative identities. If the preferred pools are insufficient, remaining candidates fill the requested count, ordered to favor nearby rate deciles and other sequences. This fallback can reintroduce candidates outside the preferred temporal/rate restrictions. Cross-sequence temporal distances use the stored sequence-time indices. Consequently, the sampler defines an empirical contrast distribution, not a guarantee of independence or complete shortcut removal. The per-query divisor $\log K_{t,h}$ in Equation 2 accounts for varying candidate counts; it is part of the implemented objective.

Optimization and selection. The main configuration uses AdamW, weight decay 0.01, batch size 128, gradient-norm clipping at 1, and up to 12,000 updates. A deterministic sequence-isolated inner split selects neural checkpoints and learning rates. The original schedule evaluates every 500 updates with learning rates 3×10^{-4} and 10^{-3}. The final motor/navigation updates evaluate every 100 updates and additionally consider 10^{-4}; the best inner-loss seed-zero learning rate is used for the three seed runs, each retaining its own inner-loss checkpoint. The original and revised patience settings correspond to six 500-update or thirty 100-update checks after the minimum training period. These changes followed development-result inspection. Thus, inner selection prevents direct use of outer scores for a checkpoint within a configuration, but does not make the complete development process an independent test or an equal-tuning-budget comparison.

CEBRA-Time. We use the official convolutional encoder, relabeling its receptive-field endpoint as the present. It outputs a 128-dimensional state from the same observed history duration. Training uses sequence-safe temporal positives at $+100$ ms and independently drawn negative contexts. This is a neural-only causal-window adaptation of CEBRA-Time, not CEBRA-Behavior or a claim about all CEBRA configurations. The official encoder and criterion are retained; the local data window and sampling construction are disclosed.

NDT2. The comparison uses the official six-layer, 256-dimensional `SpaceTimeTransformer` backbone and a two-layer causal decoder, trained from random initialization on each recording. Individual integer counts are embedded and grouped into tokens of 32 neurons, with local time/space position construction. Half of the time/space tokens are masked for Poisson infilling; an earliest-time visible token avoids fully masked causal-attention rows. Padding neurons are excluded from

the loss, and out-of-vocabulary counts are rejected rather than clipped. The online representation averages spatial tokens at the current time only. This is the official backbone with a local causal tokenization/infilling adapter, not an official pretrained checkpoint. The local adapter, native 256D representation, and distinct compute requirements limit interpretation of any ranking as a comparison of the broader model families.

C READOUTS, UNCERTAINTY, AND EVALUATION ACCESS

Future-neural likelihood. A standardized linear log-rate map is fitted by regularized Poisson regression. Feature means/scales are fitted separately within each inner training partition; standard deviations have a numerical floor. The regularizer is selected from $\{0.001, 0.01, 0.1, 1, 10\}$ using sequence-isolated calibration, and the final map is refitted on allocated training observations. Fits use float64 L-BFGS with convergence checks. For rates $\widehat{\lambda}_{t,n}$, future counts $y_{t,n}$, and reference rates λ^0_n, the reported score is

$$\mathrm{BPS} = \frac{\sum_{t,n}\left[y_{t,n}\log(\widehat{\lambda}_{t,n}/\lambda^0_n) - \widehat{\lambda}_{t,n} + \lambda^0_n\right]}{\log(2)\sum_{t,n} y_{t,n}}. \tag{4}$$

Likelihood terms and spikes are pooled over the evaluated neurons and anchors; this is not an equal-neuron average. In Figure 2, reference rates are the training future-target means at the evaluated horizon. The state-only, recent-activity-only, and concatenated readouts use the same fitting family and regularization grid but have different feature dimensions and independently fitted coefficients. An incremental gain measures accessibility under this specified family, not a nonparametric conditional-information estimate or information absent from the complete raw history.

Frozen task probes. Each nested budget allocates complete acquisition sequences at fractions 1, 5, 10, 25, and 100%. Inner calibration labels count toward that budget. Ridge and k-nearest-neighbor probes are available to all methods; visual classification additionally uses multinomial logistic regression. The neighbor grid is $k \in \{1, 4, 9, 16, 25, 49, 81, 121\}$, with uniform weighting and either native cosine or standardized Euclidean distance. The logistic grid is $C \in \{0.01, 0.1, 1, 10, 100\}$, using L-BFGS with tolerance 10^{-4} and at most 5,000 iterations. Each budget selects its family and hyperparameters on inner calibration only, then refits using all allocated labels. Family ties follow ridge, neighbors, logistic order. Original fixed linear probes remain separately reported. Budgets lacking independent calibration sequences or visual class coverage are unavailable; curves do not impute such estimates.

Regression reports the native task R^2; vision reports balanced image accuracy from the online state at 200 ms after the binned image onset. These values are not pooled into a cross-function score. For the condition-label-held-out test, only the probe's fitting labels exclude a condition. Encoder exposure is retained, so this is distinct from retraining an encoder without that neural condition.

Biological uncertainty. Technical seeds are averaged within recording before sessions are averaged within animal. A displayed animal contributes only when its required sessions, methods, and seeds are eligible for that comparison. Bootstrap intervals resample animals; they are descriptive and do not account for adaptive model development. Single-recording examples receive no population interval. The four exact paired sign-flip tests of Figure 2's incremental gains use Holm correction across functions; none crosses the adjusted 0.05 threshold. At sparse label budgets, eligible animals can differ; Figure 12 therefore displays individual animal curves, while supplementary cohort-mean curves are not fixed-cohort learning-curve estimates. Timing aggregates give equal weight to four animals, and the two paradigms are also shown separately.

Task comparisons and individual label budgets. For the full-label task analysis, we fixed a family of eight exploratory NPC contrasts before statistical testing: four functions, each compared with CEBRA-Time and NDT2. Animal-level method differences were formed after averaging all required seeds and sessions; exact two-sided paired sign-flip tests enumerated all 2^n sign patterns, and Holm correction covered the eight contrasts. Table 4 reports the observed mean differences and adjusted results. The small cohorts and prior adaptive development make the effect sizes descriptive; failure to reject does not establish equivalence.

Table 4: Exploratory full-label task contrasts. Differences are NPC minus comparator in native task units (R^2, or visual balanced accuracy). No contrast reaches the Holm-adjusted 0.05 threshold.

Function	Comparator	Mean difference	Animals	Holm p
Motor	CEBRA-Time	$+0.1858$	4	.875
Navigation	CEBRA-Time	$+0.0636$	4	.875
Vision	CEBRA-Time	$+0.1301$	6	.875
Timing	CEBRA-Time	$+0.0776$	4	.875
Motor	NDT2	-0.0462	4	.875
Navigation	NDT2	-0.0143	4	1.000
Vision	NDT2	-0.3340	6	.250
Timing	NDT2	$+0.0038$	4	1.000

At budget fractions 1, 5, 10, 25, and 100%, eligible animal counts are respectively $(0, 2, 4, 4, 4)$ for motor, $(0, 0, 1, 4, 4)$ for navigation, $(0, 0, 2, 6, 6)$ for vision, and $(4, 4, 4, 4, 4)$ for timing. Budget allocation requires enough independent sequences and fitting/calibration observations; visual probes additionally require every class in both partitions. An animal appears only if its required sessions and seeds are complete. Figure 12b therefore displays individual curves and unavailable points explicitly rather than connecting changing cohort averages. Panel (c) uses separately trained encoders and fitted readouts in each motor session: positive scores across sessions support repeated applicability of the procedure, not transfer or invariance of a fixed representation.

Illustrative readouts. Task-decoding examples use one metadata-selected recording per function, seed zero, with the full-label probe selected on inner calibration. Predictions are evaluated on the outer validation split. Motor and navigation display the median-duration eligible validation sequence; vision shows all validation predictions as a confusion matrix, and timing shows trial-mean predictions. Records are the lexicographically first eligible identifier per function: motor C (2013-10-03), Achilles, visual session 1044594870, and Cue–Set–Go animal A. These displays do not select examples by decoding score. The population-forecast examples use rolling predictions of native bins, summed into trailing 100-ms display windows (five 20-ms bins, or four 25-ms navigation bins). The motor recording remains fixed; other functions use the lower-median input-neuron-count recording and an eligible sequence of lower-median duration. These metadata rules do not inspect forecasting performance. Population-pattern panels order neurons by training mean rate and use the same linear color scale within each observed/predicted pair; scales can differ between functions. Display aggregation does not change native-bin scores.

D Random-initialization task and neural controls

Task accessibility does not by itself measure what representation learning adds. A complete descriptive screen compares random and neurally trained NPC/NDT2 encoders in seven metadata-selected recordings: two motor sessions, two rats, two mice, and one timing recording, at seed zero (14 architecture–recording cells). Random encoders use the corresponding trained-source fitting-partition normalization, output dimensions, task targets, and anchors, with no neural training. Both arms use the original frozen full-label linear probe protocol. The trained sources belong to that historical screen, rather than the final inner-selected probe menu or the entire 234-checkpoint cohort of Figure 3. All selected cells are reported in Table 5; none is omitted according to the direction of its effect.

Trained NPC has higher original linear task scores than matched random encoders in the two motor, two navigation, and one timing recordings, but lower image balanced accuracy in both visual recordings. Random NDT2 encoders already achieve 0.948 and 0.944 visual balanced accuracy; neural training increases these to 0.987 and 0.995. Thus, much of the accessible image identity can survive an untrained mapping, and neural training need not preserve its linear accessibility. These controls support neither a claim that task structure is created entirely by predictive learning nor a general conclusion about all random-feature architectures. They motivate the held-out-neuron and common-trunk objective tests in the main text.

Table 5: Original linear task readout with random and trained encoders. Scores are balanced accuracy for vision and R^2 otherwise. Each entry is one recording and initialization; no biological confidence interval is inferred.

Recording	NPC random	NPC trained	NDT2 random	NDT2 trained
Visual 1047969464	0.692	0.494	0.948	0.987
Visual 1049514117	0.739	0.676	0.944	0.995
Motor C, 2013-10-03	0.239	0.484	0.446	0.376
Motor J, 2016-04-05	0.091	0.281	0.225	0.236
Navigation Achilles	0.568	0.928	0.751	0.938
Navigation Buddy	0.268	0.479	0.248	0.457
Timing Haydn	0.122	0.393	0.282	0.418

Matched future-neural readouts of random and trained states. A separate post-hoc sensitivity analysis evaluates the same seven recordings, two architectures, and seed-zero initializations at a 400-ms neural horizon. It reuses hash-verified trained and random feature caches without updating encoders or selecting checkpoints. Eligible anchors come from the original task-probe caches and require a complete causal context and a future target in the same sequence. A recording/split-keyed deterministic subset, fixed before inspecting scores, retains at most 2,048 training and 2,000 validation anchors (2,048 training anchors in each cell; 870–2,000 validation anchors). Within each architecture–recording cell, trained and random states use identical anchors, future counts, groups, and feature dimensions. Five regularized Poisson readouts predict counts of neurons excluded from representation training: recent activity alone, trained state alone, random state alone, recent activity plus the trained state, and recent activity plus the random state. Recent activity contains current input counts and trailing 100-ms, 200-ms, and full-context means, followed by $\log(1 + x)$. Each fit selects its regularizer from $\{0.001, 0.01, 0.1, 1, 10\}$ on sequence-isolated inner calibration, with feature standardization fitted inside that split, before refitting on all allocated training anchors. Bits/spike uses the mean of the selected training future targets as its constant-rate reference. The trained-minus-random state score tests accessibility through the standalone readout; the difference between the two augmented scores tests learning beyond the same recent-history features. This is a development sensitivity analysis of original screening checkpoints, not a replacement for Figure 2 or an independent replication. All 14 cells are reported without selection by effect direction; one initialization per cell provides no population uncertainty estimate.

Table 6: Random-initialization controls at 400 ms, one seed per recording. State columns report bits/spike; Δ_R and Δ_T denote gains from adding random or trained states to identical history features. Their difference tests learning beyond history. These screening checkpoints differ from the main benchmark.

Recording	Encoder	History	Random	Trained	Δ_R	Δ_T
Visual 1047969464	NPC	0.1080	0.0210	0.1666	0.0006	0.0142
Visual 1047969464	NDT2	0.1080	0.0799	0.1452	0.0044	0.0114
Visual 1049514117	NPC	0.0661	0.0077	0.1252	0.0006	0.0151
Visual 1049514117	NDT2	0.0661	0.0474	0.1145	0.0044	0.0137
Motor C	NPC	0.0088	0.0020	0.0170	-0.0019	0.0043
Motor C	NDT2	0.0088	0.0086	0.0049	-0.0057	-0.0071
Motor J	NPC	0.1322	0.0758	0.2521	-0.0218	0.1072
Motor J	NDT2	0.1322	0.1243	0.1609	0.0233	0.0476
Achilles	NPC	0.5084	0.3341	0.6805	0.0230	0.1214
Achilles	NDT2	0.5084	0.3782	0.7425	0.0362	0.1612
Buddy	NPC	0.0796	0.0175	0.0709	-0.0015	0.0467
Buddy	NDT2	0.0796	0.0810	0.0947	0.0532	0.0572
Timing Haydn	NPC	0.0604	0.0137	0.0917	-0.0060	0.0263
Timing Haydn	NDT2	0.0604	0.0483	0.0831	-0.0012	0.0206

Trained NPC exceeds matched random encoders on both standalone and history-augmented neural prediction in all seven recordings. Trained-minus-random gains range from 0.0149 to 0.3465 bits/spike for the state alone and 0.0062 to 0.1290 after history augmentation. NDT2 improves

both endpoints in six of seven recordings; the Motor C effects are negative (-0.0037 and -0.0013 bits/spike, respectively). In the two visual recordings, NPC's standalone neural gains are 0.1457 and 0.1175 bits/spike, and its augmented gains are 0.0136 and 0.0144, despite the lower trained linear image accuracy in Table 5. This demonstrates a difference in readout accessibility under the tested learning intervention, not a change in total neural or task information. No pooled significance test or population-level superiority claim is inferred from this small, post-hoc screen.

E COMMON-TRUNK OBJECTIVE EXPERIMENT

This completed diagnostic includes MC_RTT, the rat Buddy, two visual development recordings, and one DMFC recording. Six variants and three technical seeds yield 90 runs. Within a recording and seed, variants share the initial inference trunk, neural partitions, context windows, query anchors, minibatch schedule, and 4,000 AdamW updates at learning rate 3×10^{-4}. The final-update checkpoint is evaluated. Visual states are 128D and other states are 32D; per-bin targets are 16D with a 64D hidden encoder. Temperature is 0.1, with at most 255 negatives. These settings differ from the final main benchmark and are not presented as ablations of every final encoder variant.

The five native horizons are 20/40/80/160/320 ms, except navigation's 25/50/100/200/400 ms. Future likelihood is averaged over these horizons within each recording and seed. Task probes use full-label ridge classification for vision and regression otherwise. The internal temporal control retains causal input, uses symmetric one-bin target offsets, and shares its predictor across offsets. Thus, it changes target offsets and head tying and does not isolate time direction alone. The count-prediction control trains horizon-specific Poisson heads on input-neuron counts. Loss-specific heads and training times differ; equal update counts do not mean equal compute or optimization maturity.

Table 7: Complete objective controls. Entries average three technical seeds within each recording. Task scores are balanced accuracy for vision and R^2 otherwise; rows are recordings, not independent cohorts.

Recording	NPC	Temporal	Count	Shuffled	One bin	Global
Future-neural prediction (bits/spike)						
MC_RTT	0.117	0.073	0.066	0.036	0.114	0.096
Buddy	0.261	0.179	0.105	0.120	0.232	0.222
Visual 1	0.387	0.259	0.305	0.064	0.341	0.409
Visual 2	0.256	0.157	0.180	0.050	0.227	0.277
DMFC	0.101	0.062	0.071	0.013	0.092	0.104
Task score						
MC_RTT	0.367	0.137	0.160	0.034	0.369	0.344
Buddy	0.467	0.271	0.287	0.127	0.312	0.407
Visual 1	0.288	0.185	0.196	0.139	0.197	0.266
Visual 2	0.366	0.232	0.175	0.142	0.247	0.339
DMFC	0.229	0.104	0.166	0.084	0.206	0.251

All five recording-level means favor NPC over the temporal and count controls on both endpoints. The shortest-horizon comparison favors multihorizon training for future prediction in all five recordings, but MC_RTT task readout decreases slightly. Global negatives increase visual future-neural likelihood and both DMFC endpoints relative to structured sampling, while reducing motor/navigation means and visual task accuracy. The shuffled-future control reduces both mean endpoints in all five recordings but retains nonzero utility. This is expected: a contrastively trained causal encoder can remain informative even when a particular future association is disrupted. A paired sensitivity analysis retains every seed and horizon rather than only recording means. For each of the temporal, count-prediction, and shuffled-target controls, NPC improves both endpoints in all 15 recording–seed pairs and neural prediction in all 75 recording–seed–horizon comparisons. Multihorizon training improves horizon-averaged neural prediction in all 15 seed pairs, but only 63 of 75 individual horizon comparisons; its task advantage holds in 14 of 15 seed pairs. These counts assess sensitivity to averaging, not independent sample size or a pooled hypothesis test. NPC/count recorded training-time ratios range from 20.7 to 29.4 across recording–seed pairs; runtimes include the measured training

procedure and are not a hardware-normalized compute measure. No between-animal confidence interval is inferred from the three seeds.

F FITTED EVOLUTION AND ITS CONTROLS

States remain in physical time, and missing bins or sequence boundaries break trajectories. A sequence-weighted PCA fit on training states retains 16 dimensions with deterministic signs and one global scale, without whitening. Each grouped inner fold refits this preprocessing. Affine transitions are ridge fitted separately at 100, 200, and 400 ms, using three sequence-isolated folds and regularizers from 0.001 through 1000 in powers of ten. These are direct horizon-specific forecasts, not iterated autonomous rollouts. Sequence-balanced displacement skill compares transition error against persistence; negligible displacement variance yields an unavailable estimate.

A concurrent Poisson emission map is fitted from the current state to the concurrently measured reserved neurons, with frozen scaling and coefficients. At evaluation, the same map reads out the static state and its evolved counterpart against the identical future neural counts. The transition fit never receives those reserved-neuron targets. Figure 4's constant-rate reference is the sequence-weighted concurrent-training count mean, unlike Figure 2's horizon-specific future-training mean. The direct-history comparator is fitted directly to future counts. Consequently, absolute bits/spike from Figures 2 and 4 are not interchangeable benchmark measurements even when their units agree.

The task/time reference diagnostic uses training condition/event-time means in 100-ms bins to predict the future latent state. History references reuse the current and trailing-mean neural feature definitions, with the dynamics study's declared preprocessing. In the six-mouse quantitative cohort, the 400-ms NPC task/time reference averages -0.2124: one mouse scores -2.9835, making the distribution strongly heterogeneous. Skill relative to persistence is unbounded below; the complete values for all six animals are retained. This heterogeneous result does not establish an absence of task-related structure.

In the matched main cohort, NPC's motor 400-ms displacement skill is 0.5548 versus 0.6376 for the task/time reference; the reference is higher in all four animal means. Navigation means are 0.4629 and 0.1213; timing means are 0.4409 and -0.4069. These references retain all animals, including negative skill values. Context windows overlap at 100 and 200 ms, and navigation's 500-ms context also overlaps at 400 ms. Thus, positive displacement skill and visually structured trajectories alone cannot distinguish trial-specific dynamics from shared observations or task progression. Large evolved-minus-static neural gains can also result partly from a poorly calibrated static readout; absolute scores and direct-history references are therefore retained.

G ADAPTATION TO SOURCE-DISJOINT VISUAL MICE

Sources, targets, and training. The visual transfer protocol uses five source sessions from three mice and sixteen target sessions from eight different mice, with two target sessions per mouse. Source and target animal identifiers are disjoint. The target cohort is a previously inspected validation split; no independent test mice are used here. Three source seeds receive 5,000 neural-pretraining updates. Each target input interface then receives 250 updates with the source core frozen. Target-only controls train the representation from random initialization for either 250 or 5,250 updates. The latter matches the source-plus-adaptation update count, but not data exposure, trainable parameters, or compute. The archived source protocol uses a 128-dimensional representation and neural forecasting horizons of 20, 40, 80, 160, and 320 ms, distinct from the main benchmark.

Budgets and readout pairing. Positive-query fractions are 1, 5, 10, 25, 50, and 100% of eligible target-training anchors. Only positive query anchors are restricted. Negative candidates are drawn from the full target-training index, and population-rate deciles are computed from that full training recording in every arm. The fractions therefore do not limit all neural observations accessed during encoder adaptation. At every budget, the Poisson readout uses 8,192 anchors sampled from the full target-training partition and evaluates 12,000 validation anchors. The same target neurons, query times, future observations, and readout procedure are used across arms. Training and validation sequence groups are disjoint. For the 10% and 25% pretrained arms, 96 archived readouts used a different anchor construction; we recomputed those readouts from the frozen encoders on the control

anchors before aggregation. No representation was retrained for this repair. All 864 arm–budget–session–seed cells are complete.

Aggregation and interpretation. Scores average the five horizons, then seeds and sessions within mouse, with equal weight for the eight mice. Intervals use 20,000 paired mouse-bootstrap samples. Two-sided exact sign-flip tests are Holm corrected across six budgets separately for each target-only control. The abstract's 7% improvement uses full query coverage and compares pretrained adaptation with the 250-update target-only control. Figure 5b shows all eight target mice, averaging three seeds and two sessions per mouse. Mouse indices 1–8 correspond to identifiers 524925, 530862, 533537, 536480, 544838, 572846, 574078, and 578003. A separate seed-sensitivity analysis retains every source-training seed. Mean gains are 0.0274, 0.0142, and 0.0239 bits/spike for seeds 0, 1, and 2; respectively eight, five, and seven of eight mice improve. These seeds assess training sensitivity within the same mice, rather than additional biological replication. The 12% improvement at the 1% query fraction is the ratio of animal-mean bits/spike scores, $0.225/0.200 - 1$, against that same control. The corresponding ratio against the 5,250-update control is 49%. These percentages describe mean likelihood-gain scores, not forecast-error reduction. Unlike the main within-recording benchmark, this historical readout computes its constant-rate reference from the validation mean of each target neuron. All adaptation arms share that reference, but its absolute scores and score ratios cannot be pooled with the main benchmark. The 1% emphasis is post-hoc: the original transfer decision focused on 10% and 25% budgets and did not meet its success criterion. The present reanalysis preserves those unfavorable comparisons and does not revise that earlier decision. Archived area-under-learning-curve summaries had unmatched budget ranges, and archived event-label scores included states beyond image identity; neither endpoint supports a claim here. The findings identify adaptation behavior under restricted positive-query sampling; full-recording negative candidates and fixed readout calibration prevent interpreting this as target-data efficiency.

H ADDITIONAL FIGURES

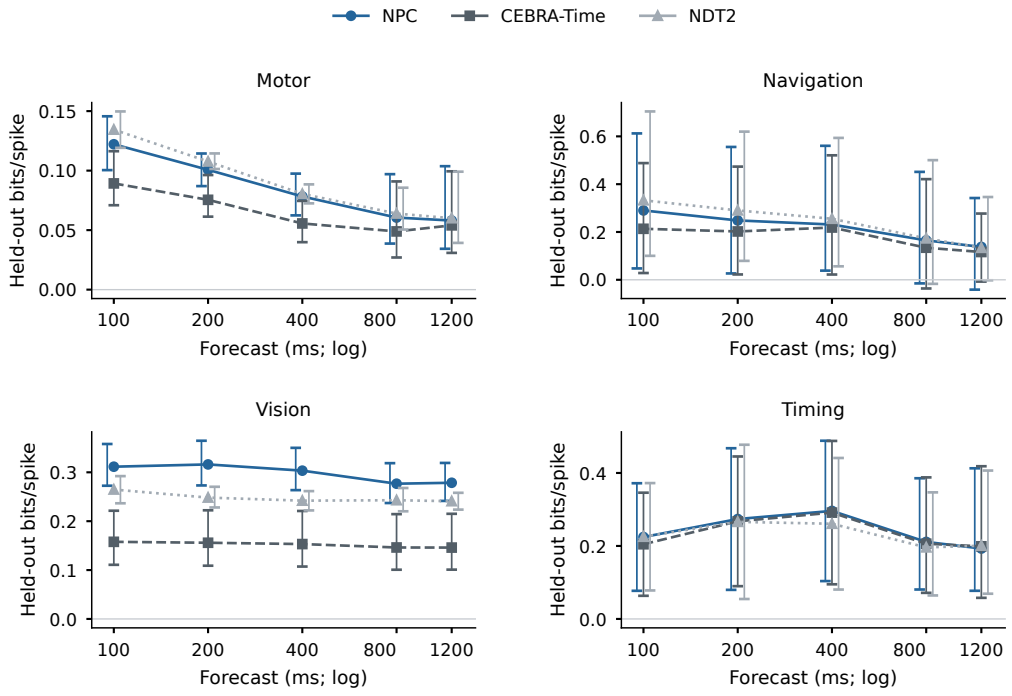


Figure 6: Full-horizon uncertainty for Figure 2a. Means and 95% animal-bootstrap intervals use the same cohort as the main figure.

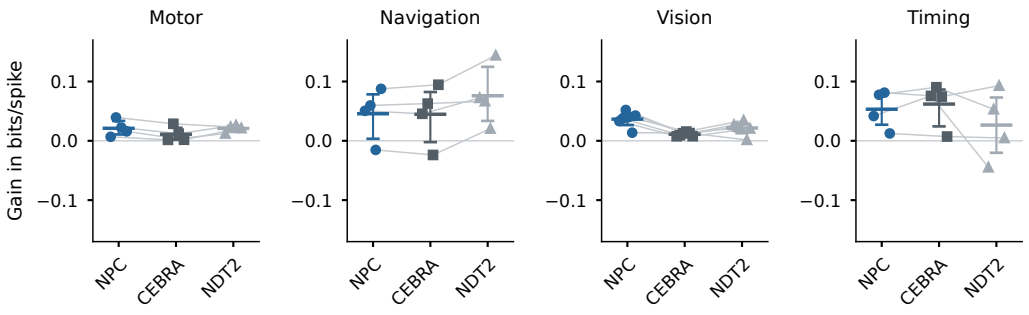


Figure 7: Incremental neural prediction at 400 ms for NPC, causal CEBRA-Time, and NDT2 when each is added to the same recent-activity summaries. Points pair the same animals as Figure 2b; whiskers show 95% animal-bootstrap intervals.

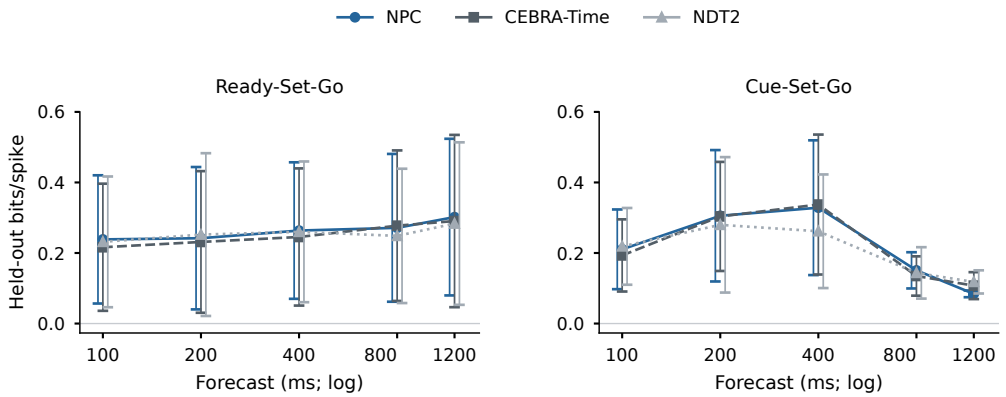


Figure 8: Future-neural prediction separately for Ready–Set–Go (Haydn/Gershwin) and Cue–Set–Go (A/D). Each paradigm has two animals; whiskers show descriptive 95% animal-bootstrap intervals.

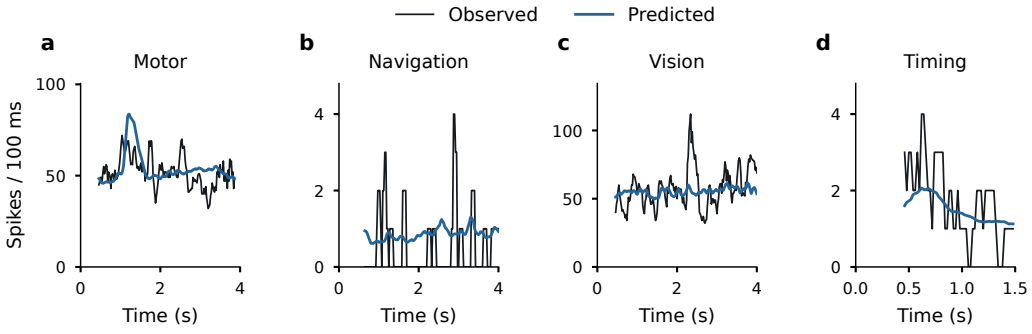


Figure 9: Rolling 100-ms-ahead population forecasts. Black denotes observed counts and blue expected counts, displayed as trailing 100-ms sums. Each prediction uses a newly observed causal context; this is not an autonomous rollout. Examples follow the metadata-based selection rule in Appendix C.

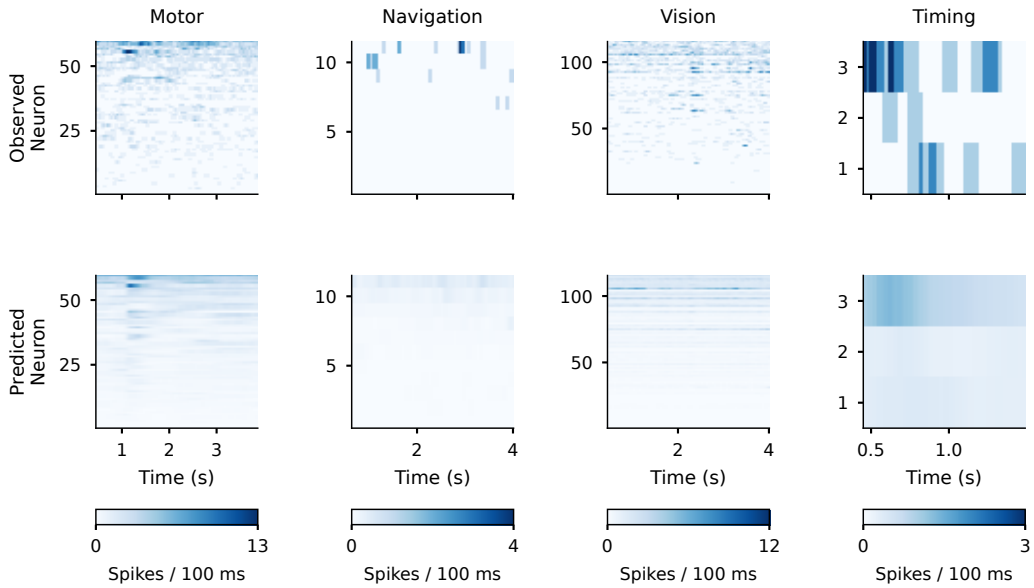


Figure 10: Observed and predicted count patterns for the same illustrative sequences. Neurons are ordered by training mean firing rate, identically within an observed/predicted pair. Each function has a shared linear color scale across the pair; functions use different scales. Counts are displayed as trailing 100-ms sums.

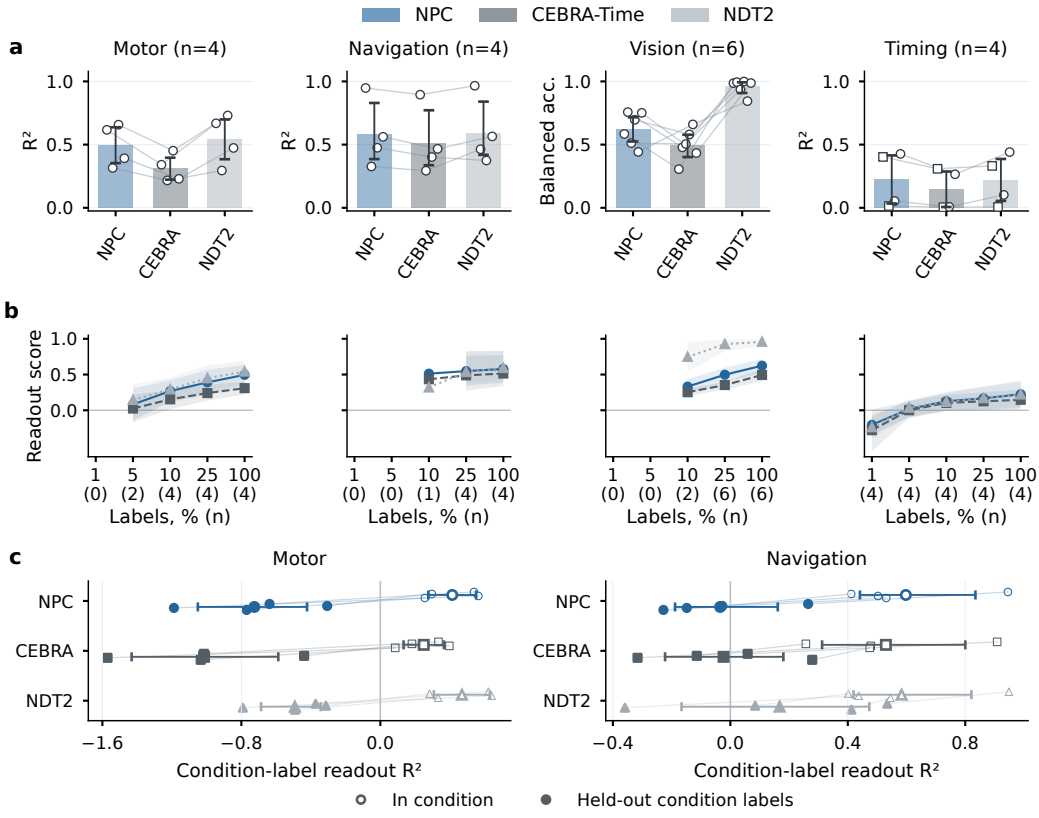


Figure 11: Complete task-readout comparisons. (a) Full-label scores; (b) sequence-label budgets; (c) condition-label-held-out motor/navigation probes. Means and 95% animal-bootstrap intervals summarize eligible animals, whose counts can change across budgets. Missing estimates are not imputed. The encoder may observe the unlabeled held-out condition; panel (c) tests readout-label generalization only.

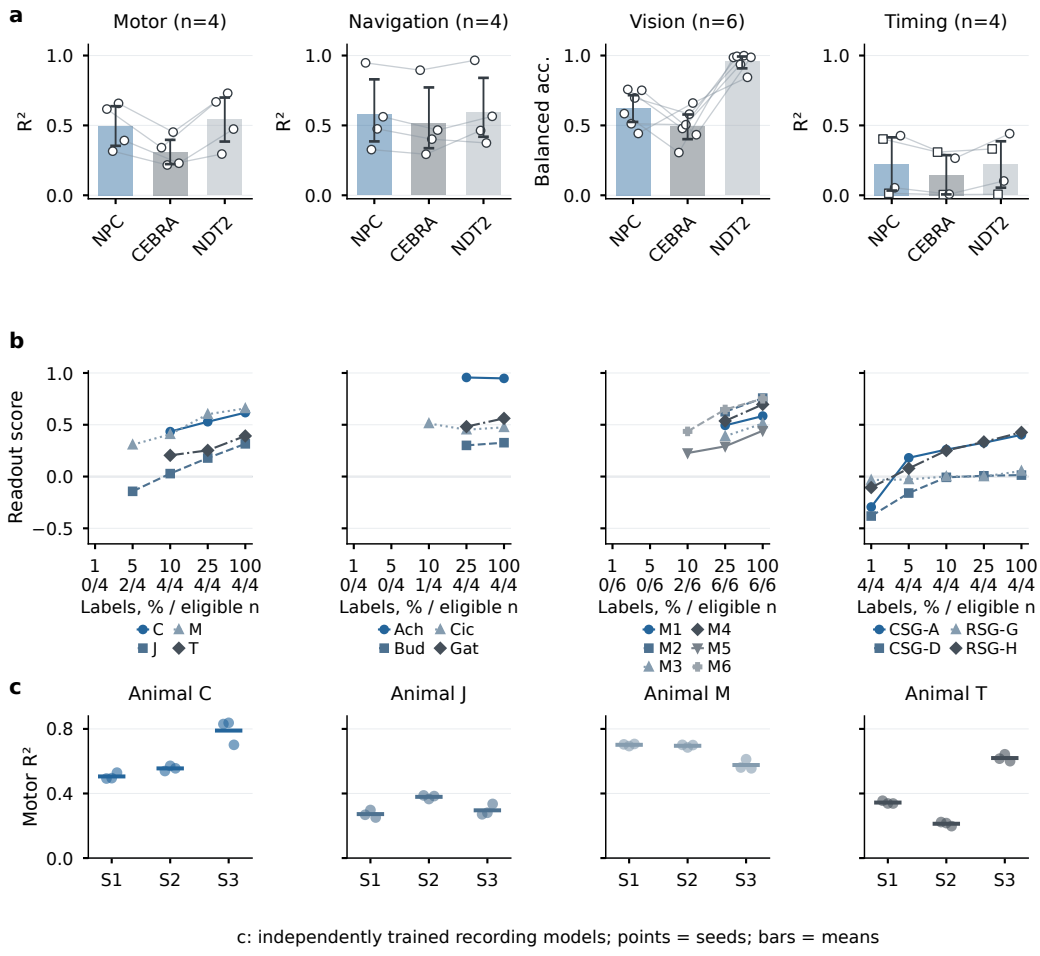


Figure 12: Individual label budgets and motor sessions. (a) Full-label method comparison. (b) NPC curves for each animal, with all five requested fractions and eligible/total counts. (c) Twelve motor sessions grouped by animal; points are technical seeds and bars their means. Each recording has an independently trained encoder and readout.

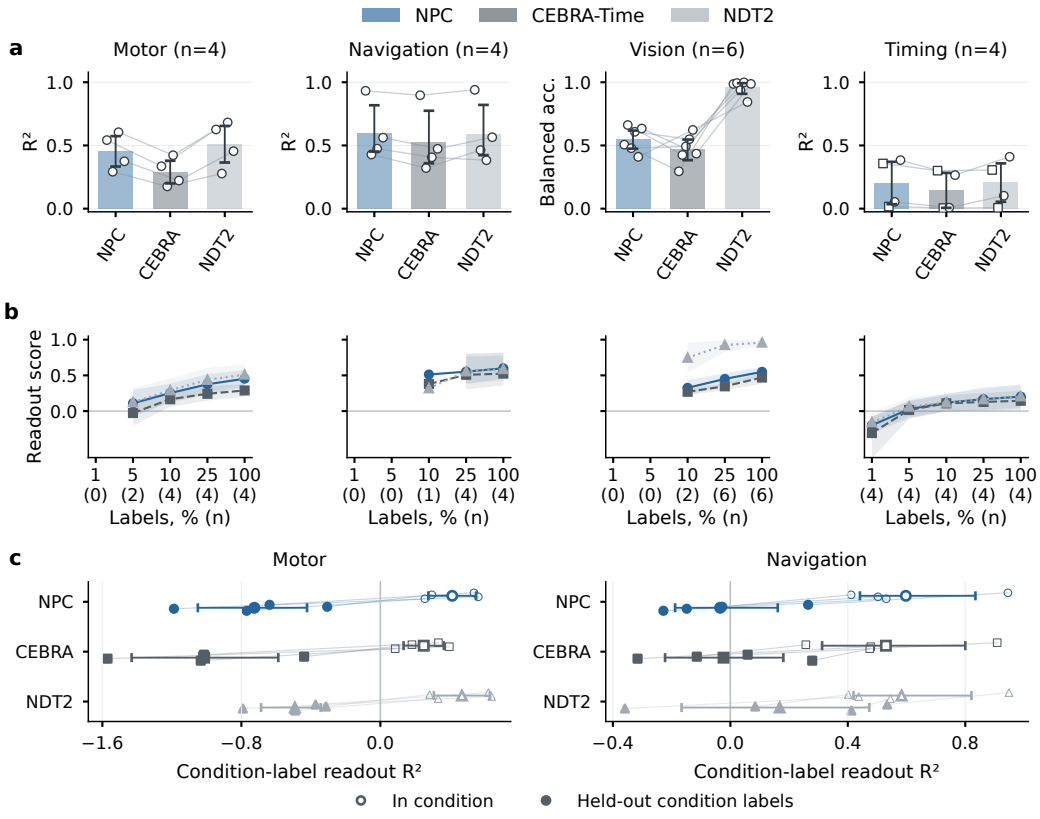


Figure 13: Fixed linear probes retained alongside the inner-selected probe family in Figure 3. Comparisons use the same encoder checkpoints and matched label budgets.

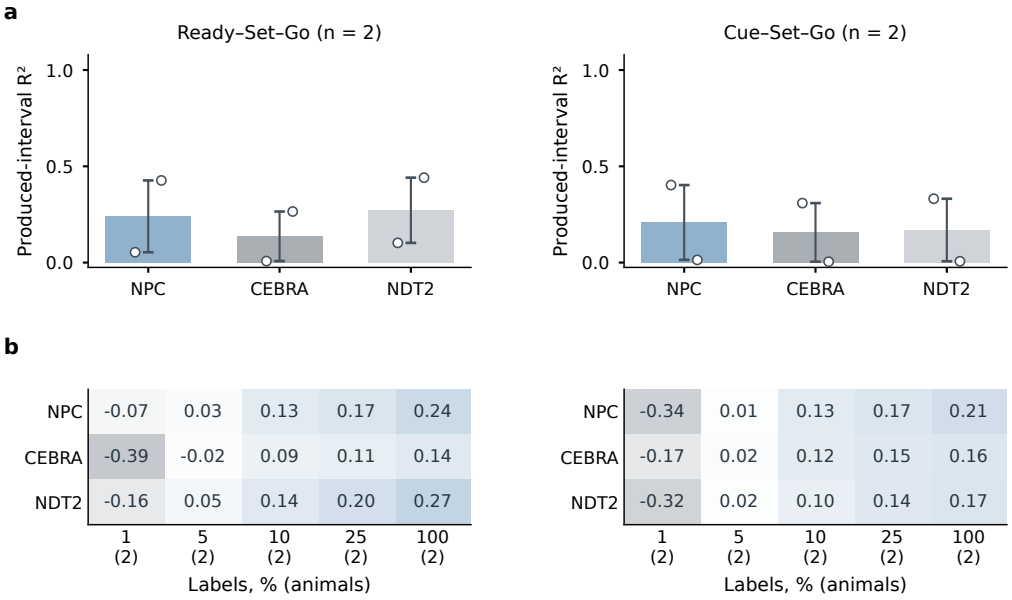


Figure 14: Task readout by timing paradigm. Each paradigm contributes two animals using the final checkpoints and probe rules from Figure 3. NPC/NDT2 mean R^2 is 0.240/0.272 in Ready–Set–Go and 0.208/0.169 in Cue–Set–Go. Whiskers show descriptive 95% animal-bootstrap intervals.

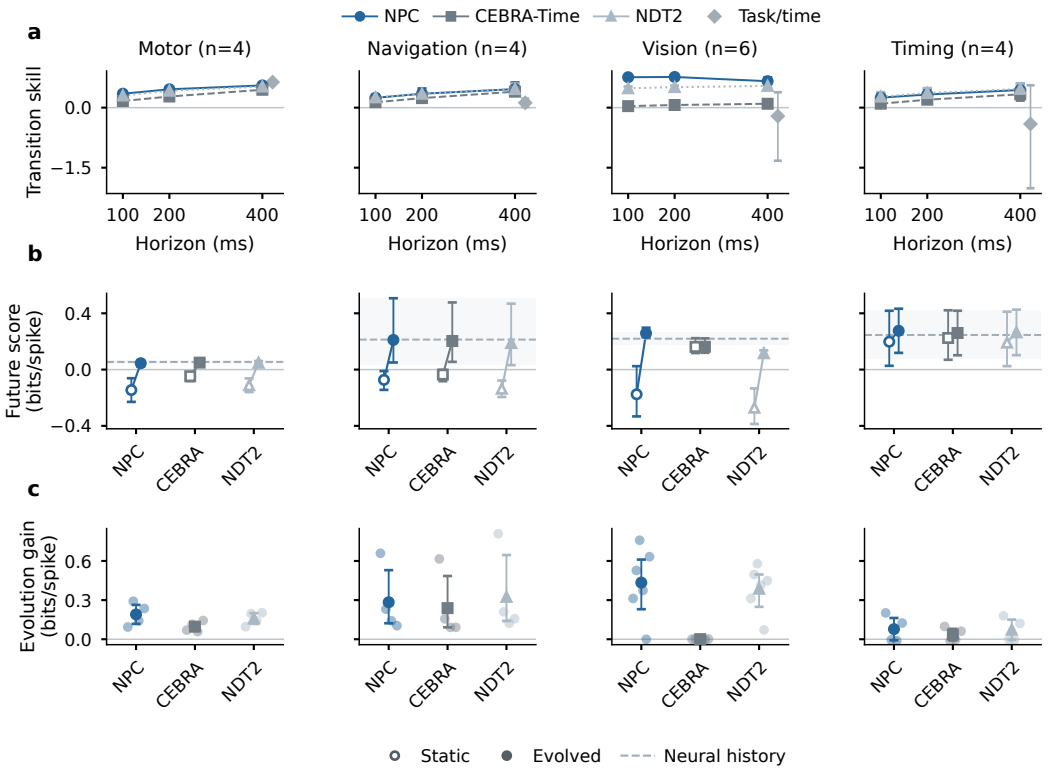


Figure 15: Complete fitted-evolution comparisons. (a) Latent transition skill across horizons; diamonds mark NPC's task/time reference. (b) Static/evolved neural forecasts and history reference. (c) Paired evolution gains. Large points are animal means; whiskers and bands show 95% animal-bootstrap intervals. All 234 main-benchmark checkpoints are included, with four timing animals.