Flow Matching with In-Context Priors for Out-of-Distribution Brain Dynamics

Sam Gijsen* **Michał Łukomski** **Marc-André Schulz** **Kerstin Ritter**

Hertie Institute for AI in Brain Health, University of Tübingen, Germany
Tübingen AI Center, University of Tübingen, Germany
Charité – Universitätsmedizin Berlin, Department of Psychiatry and Psychotherapy, Germany
German Center for Mental Health (DZPG), partner site Tübingen, Germany

Abstract

Flow matching and diffusion models enable conditional generation across domains ranging from images to proteins, with recent extensions to out-of-distribution contexts. Yet generative models of neural time series have largely remained restricted to categorical conditioning, precluding compositional and zero-shot generalization. In this work, we propose PINC-Flow, a per-timestep conditioned diffusion transformer for generating realistic fMRI brain dynamics during unseen cognitive tasks by injecting both compositional language and optional spatial priors in-context. Such zero-shot generation could enable counterfactual neuroscience by supporting in-silico design and evaluation of novel cognitive experiments before empirical validation. Leveraging this model, we evaluate across hundreds of held-out task conditions and characterize predictive performance in relation to the training manifold. From language alone, the model recovers region-specific recruitment across tasks and held-out spatial activation patterns. Spatial priors, when available, complement the text pathway by anchoring generation in regions of task space where language alone degrades, while retaining the compositional structure needed for counterfactual task specification. To our knowledge this is the first generative model of whole-cortex fMRI dynamics for unseen cognitive tasks, advancing counterfactual neuroscience and data-driven experimental design.

1 Introduction

Generative models trained to map a noise distribution to a target data distribution produce realistic samples across many modalities [Sohl-Dickstein et al., 2015, Lipman et al., 2022], including images [Rombach et al., 2022], audio [Liu et al., 2023], molecules [Hoogeboom et al., 2022], and proteins [Watson et al., 2023]. Much of the progress in conditional generation has been driven by conditioning on high-dimensional, continuous signals. Language model embeddings have proven especially effective: they provide a structured semantic space that supports compositional generalization [Radford et al., 2021, Saharia et al., 2022, Liu et al., 2023], while also enabling novel conditioning signals to be specified at inference time. For biological time series, however, it remains unclear how such conditioning can support extrapolation to experimental conditions that were never observed during training.

Such generalization is particularly relevant for biological data: acquisition is costly and candidate interventions vastly outnumber measured examples. The promise is clearly evident in molecular and protein design, where generative models produce realistic samples even away from the direct training support [Lee et al., 2023, Watson et al., 2023]. Yet, generative models of neural time series have largely remained restricted to discrete categorical conditioning such as sex, disease labels, or

*samgijsen@gmail.com • Code and models: https://github.com/SamGijsen/pinc-flows

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

experimental task identity, which prevents the flexible conditioning that enabled generalization in other domains.

We focus on functional MRI, which non-invasively measures a proxy of neural activity [Logothetis et al., 2001]. The dominant paradigm is task-based: specifically, participants perform structured cognitive experiments (e.g., viewing emotional faces, recalling memories, responding via button presses) while their brain dynamics are recorded over time [Barch et al., 2013]. Each experiment is a temporally structured composition of stimuli, instructions, and motor responses. Modeling such experiments as discrete categories conflates this compositional structure into prespecified and fixed labels, foreclosing in-silico investigation of novel cognitive tasks and counterfactual interventions. Task-based fMRI is also acutely affected by data scarcity, as each new study requires expensive scanner time [Szucs and Ioannidis, 2020].

No existing approaches address the generation of realistic whole-cortex fMRI dynamics for out-of-distribution cognitive tasks that were never observed during training. In particular, encoding models yield deterministic point estimates of group-mean responses to stimuli, but do not provide temporal dynamics, individual variability, and no native mechanism for novel-task extrapolation. Meanwhile, generative models with categorical conditioning produce realistic dynamics but cannot express experimental structure beyond their fixed label set [Vetter et al., 2024, Seo et al., 2025, Hu et al., 2025]. What is missing is a generative model that produces realistic neural dynamics and extrapolates compositionally across experiments.

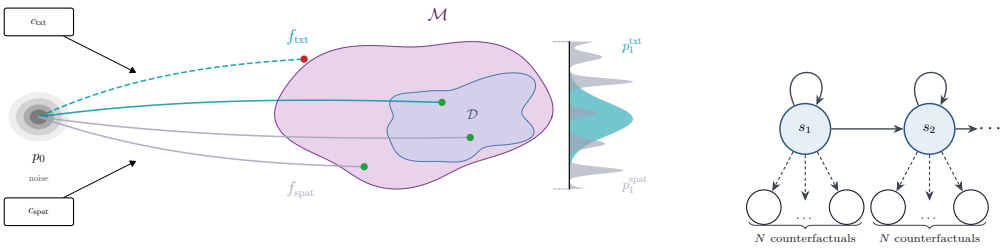
Here we present such a model: a per-timestep conditioned diffusion transformer trained with a flow matching objective under two complementary conditioning regimes (PINC-Flow; Priors-IN-Context). First, we inject compositional language embeddings of each timestep's stimulus, instruction, and response, enabling local and compositional generalization across experimental conditions. We characterize zero-shot generalization across task-space, finding strong text-only generation, albeit breaking down predictably with distance from the training support. Second, we simultaneously train the model to use in-context spatial priors via a self-supervised strategy. We show that spatial priors anchor generation in regions of task space where language alone degrades, enabling realistic data generation for any experimental design where activation patterns can be specified, while retaining the compositional structure needed for counterfactual task specification. Together, these pathways provide a first step toward generative modeling of whole-cortex fMRI dynamics for unseen cognitive tasks.

2 Related Work

Generative models of brain data. Prior work has generated functional connectivity matrices, task-based activation maps, or other aggregate neuroimaging targets rather than full neural time series [Tavor et al., 2016, Gal et al., 2022, Serin et al., 2025]. Recent diffusion models generate brain dynamics more directly [Zhai et al., 2025, Hu et al., 2025, Seo et al., 2025, Xia et al., 2026, Tew et al., 2025, Gao et al., 2026], but typically condition on fixed metadata such as demographics, diagnosis, or task identity, if at all. These models can synthesize realistic samples within observed labels, but cannot specify unseen tasks compositionally at inference time.

Continuous conditioning in neuroimaging. Encoding models predict brain responses from stimulus features, spanning hand-engineered methods [Naselaris et al., 2011] to pretrained language and vision representations (TRIBEv2, d'Ascoli et al. [2026]; Appendix N). Pretrained representations are used by decoding models to infer or reconstruct observed stimuli [Ozcelik and VanRullen, 2023, Scotti et al., 2023, Bosch et al., 2025] or in training foundation models to aid generalization [Gijsen and Ritter, 2025, Wei et al., 2025]. While these works show that continuous representation spaces can support generalization, they target deterministic prediction, stimulus reconstruction, or representation learning rather than diverse generation of task-fMRI dynamics under novel experimental designs.

Structured conditioning for biological extrapolation. Structured conditioning has enabled useful extrapolation in biological domains including protein design [Watson et al., 2023, Ingraham et al., 2023], molecular generation [Igashov et al., 2024, Schneuing et al., 2024], and single-cell perturbation modeling [Lotfollahi et al., 2019, 2023]. These settings share the central pressure of task fMRI: experiment spaces are combinatorial, acquisition is expensive, and directly observed conditions sparsely cover the space of scientific interest.



(a) (b)

Figure 1: a) The flow model is trained to enable two conditional sampling functions f_{txt} and f_{spat}, with distinct induced densities. Solid paths denote intended transports to valid trajectories on $\mathcal{M}$; the dashed path illustrates a possible failure mode under unsupported text conditioning. b) To alleviate overfitting to sequence ordering during training, we occasionally present counterfactual sequences.

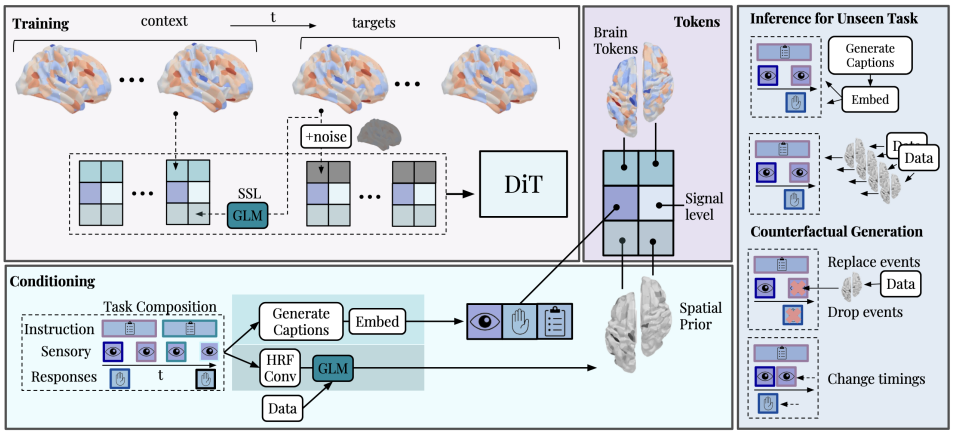


Figure 2: We train a diffusion transformer (DiT) to sample trajectories of brain dynamics. The model is conditioned on context volumes, text embeddings of task captions, and spatial priors. Both conditioning pathways enable compositional generalization by decomposing experimental tasks.

3 Methods

We aim to model the distribution of brain dynamics that a participant would exhibit during any plausible cognitive experiment. Let $\mathcal{M}$ denote the manifold of valid task-fMRI trajectories induced by the space of such experiments, and let $\mathcal{D}_{\text{train}} \subset \mathcal{M}$ denote the finite collection of trajectories captured by our training data. Although $\mathcal{D}_{\text{train}}$ covers only a small portion of $\mathcal{M}$, two structural properties make generalization beyond it tractable. First, experimental conditions are compositional: each is a structured combination of stimuli, instructions, and motor responses, components that recur across experiments and enable natural language descriptions. Second, both the temporal and spatial signature of activation for a given cognitive operation have a significant shared structure between studies and subjects. We exploit both properties through a dual conditioning scheme depicted in Figure 1a. A text-guided sampler f_{txt} uses the compositional structure of natural-language descriptions, supporting local and compositional extrapolation across $\mathcal{M}$. However, when queried with unsupported or far-extrapolative text conditions, f_{txt} may fail to extrapolate within $\mathcal{M}$, instead producing trajectories that leave the manifold of valid task-fMRI dynamics. A spatially guided sampler f_{spat} instead conditions on an explicit activation map, helping to anchor generation within a constrained region of $\mathcal{M}$ while requiring the activation pattern to be provided or estimated externally. Together, the two pathways trade compositional flexibility against distributional anchoring and provide complementary coverage of $\mathcal{M}$.

We instantiate the above as a generative model of fMRI dynamics with three design choices. First, whereas prior work conditions diffusion models through global mechanisms such as adaptive layer-norm, which do not allow per-timestep resolution, we instead insert conditioning tokens directly into the transformer context. This allows the cognitive content of each timestep to be specified

independently and enables counterfactual interventions on event timing and composition. Second, both conditioning pathways operate in continuous, structured spaces, making local and compositional generalization possible at inference. Third, we train the model end-to-end with a flow matching objective, yielding realistic and diverse trajectories rather than deterministic point predictions. We detail the architecture and training procedure below.

3.1 Generative Model

We model parcellated fMRI sequences. At each timestep t, $\mathbf{x}_t \in \mathbb{R}^P$ represents the activation of all P parcels, with $x_{t,p}$ denoting parcel p. We split $\mathbf{x}_t$ into two hemispheric vectors $\mathbf{x}_t^h \in \mathbb{R}^{P/2}$ for $h \in \{L, R\}$. A training sample spans $T = T_c + T_g$ timesteps, with T_c real-data context volumes and T_g generation volumes. The latter are noised during training and iteratively denoised at inference. Hemisphere-specific encoders E_L, E_R project $\mathbf{x}_t^h \mapsto \mathbf{z}_t^h \in \mathbb{R}^D$, whereas the decoders $\mathrm{Dec}_L, \mathrm{Dec}_R$ map back to parcel space.

Architecture. We use a Diffusion Transformer [Peebles and Xie, 2023] over the flat token sequence of length $6T$ consisting of six tokens per timestep:

$$\left\{(\mathbf{z}_t^L,\, \mathbf{z}_t^R,\, \mathbf{c}_t^{\text{spat},L},\, \mathbf{c}_t^{\text{spat},R},\, \mathbf{c}_t^{\text{txt}},\, \mathbf{c}_t^{\sigma})\right\}_{t=1}^{T}$$

to which we add a learnable temporal embedding $\mathbf{e}_t$ and token-type embedding. The *brain tokens* $\mathbf{z}_t^h$ are the hemisphere-encoded activations for $t \le T_c$, and the noised target for $t > T_c$. The *spatial-prior tokens* $\mathbf{c}_t^{\text{spat},h} = E_h(\boldsymbol{\beta}_t^h)$ project a per-timestep spatial prior map $\boldsymbol{\beta}_t^h \in \mathbb{R}^{P/2}$ (defined below) through the same hemispheric encoder used for brain tokens; this shared projection ensures that brain-state and spatial-prior tokens occupy a common representational space.

The *language token* $\mathbf{c}_t^{\text{txt}}$ is constructed compositionally from three captions describing the cognitive content at t: a sensory caption (e.g., "visual: photos of human faces"), an instruction caption ("choose the most attractive option"), and a response caption ("right thumb button press"). Each caption is embedded with a pretrained sentence transformer [Xiao et al., 2023], linearly projected to dimensions $0.4D$, $0.4D$, $0.2D$, and concatenated to form a single language token in $\mathbb{R}^D$. By embedding the three components independently, the language token decomposes the experimental description along its natural axes of variation, allowing recombinations unseen during training. We randomly mask components during training. To improve robustness to specific phrasings, we generate six phrase variants per caption with a language model (Appendix P) and sample randomly. Finally, the *signal-level token* $\mathbf{c}_t^{\sigma}$ embeds the scalar signal level σ_t. To enable classifier-free guidance at inference, language tokens, spatial-prior tokens, and context volumes are independently dropped during training.

We condition on natural-language descriptions rather than modality-specific stimulus encoders (e.g., CLIP image embeddings, audio encoders). This unifies the sensory, instruction, and response components of a task in a single embedding space and scales to datasets in which raw stimulus files are unavailable (e.g. the Nakai dataset introduced below).

Self-supervised spatial prior. The spatial-prior pathway is trained self-supervised, exploiting a fundamental property of the fMRI signal. Neural activity drives a slow hemodynamic response function $hrf(t)$, well-modeled as a canonical double-gamma kernel peaking at ≈ 5 s and decaying over ≈ 15 s [Friston et al., 1998]. The general linear model (GLM) routinely used in fMRI analysis models the signal as

$$x_{t,p} \;=\; \sum_k \beta_{p,k}\,(hrf * s_k)(t) \;+\; \varepsilon_{t,p},$$

where $s_k(t)$ is the onset indicator of event type k and $\beta_{p,k}$ is a per-parcel regression coefficient. Because the HRF spreads each event's effect over the upcoming temporal window, the event coefficient encodes the spatial pattern expected to modulate the upcoming response window. Concretely, for an event of type k with an onset at timestep t, we refer to the coefficients $\{\boldsymbol{\beta}_{p,k}\}$ over parcels p belonging to hemisphere h as the vector $\beta_t^h \in \mathbb{R}^{P/2}$. To train f_{spat}, we supply $\boldsymbol{\beta}_t^h$ to the model at the matching timestep, and the GLM is fit on the subject's own run. Because the prior is derived from the same data the model reconstructs, no external labels are required, and learning to use the prior reduces to a well-posed self-supervised objective.

We estimate $\boldsymbol{\beta}_t^h$ per event type rather than jointly across events, facilitating counterfactual edits to the event composition at inference. To alleviate overfitting to specific task signatures and to promote

sample diversity, we estimate $\boldsymbol{\beta}_t^h$ per subject rather than per task. At inference, the spatial prior can be dropped, predicted from text, or supplied from external empirical sources such as prior task-fMRI studies. In our evaluation, priors estimated from held-out recordings define an oracle empirical setting. This setting is not a zero-resource prediction benchmark like f_{txt}, but tests the intended use of f_{spat}. Namely, spatially anchored generation without retraining, where supplied or recombined activation patterns are realized as realistic temporal dynamics under event, response, and timing compositions unseen during training.

Flow matching. Let $z = \{\mathbf{z}_t^h : h \in \{L, R\}, t > T_c\}$ denote the collection of brain tokens at the generation timesteps, and let $c^{\text{ctx}} = \{\mathbf{z}_t^h\}_{t \le T_c}$ denote the context volumes. We group the conditioning as $c = (c^{\text{txt}}, c^{\text{spat}}, c^{\text{ctx}})$. We learn a velocity field $v_\theta(z_\tau, \tau, c)$ that transports $p_0 = \mathcal{N}(0, I)$ to a conditional data distribution $p_1(\cdot \mid c)$ along the linear probability path $z_\tau = (1-\tau)\, z_0 + \tau\, z_1$ with FM time $\tau \in [0, 1]$ [Lipman et al., 2022]. The network uses an endpoint-prediction parameterization: it predicts $\hat{z}_1 = g_\theta(z_\tau, \tau, c)$, and the velocity is computed as $v_\theta = (\hat{z}_1 - z_\tau)/(1-\tau)$. Training draws $\tau \sim \mathcal{U}(0, 1)$, $z_0 \sim p_0$, and minimizes $\|\hat{z}_1 - z_1\|^2$ on generation timesteps. At inference we generate samples by K-step Euler integration, $z_{\tau+\Delta\tau} = z_\tau + \Delta\tau\, v_\theta(z_\tau, \tau, c)$ with $\Delta\tau = 1/K$. Appendix H describes a stochastic variant used only for calibration analyses and Appendix E ablates the endpoint parameterization.

Conditional samplers. The two pathways introduced above correspond to running the learned dynamics under different conditioning sets, inducing distinct approximate transports from p_0 to pathway-specific conditional endpoint distributions:

$$f_{\text{txt}} : p_0 \to p_\theta\big(z_1 \mid c^{\text{txt}}, c^{\text{ctx}}\big) \approx p_1^{\text{txt}}(z_1) := p_1\big(z_1 \mid c^{\text{txt}}, c^{\text{ctx}}\big)\,,$$
$$f_{\text{spat}} : p_0 \to p_\theta\big(z_1 \mid c^{\text{txt}}, c^{\text{spat}}, c^{\text{ctx}}\big) \approx p_1^{\text{spat}}(z_1) := p_1\big(z_1 \mid c^{\text{txt}}, c^{\text{spat}}, c^{\text{ctx}}\big)\,.$$

f_{txt} relies on compositional language conditioning alone; f_{spat} additionally anchors generation to an explicit activation map.

Classifier-free guidance. Independent dropout of language, spatial-prior, and context tokens during training allows v_θ to be evaluated using arbitrary subsets of c [Ho and Salimans, 2022]. At inference we group the conditioning into two channels: a *task* channel c^{task}, which is c^{txt} under f_{txt} and $(c^{\text{txt}}, c^{\text{spat}})$ under f_{spat}, and the *context* channel c^{ctx}. When a channel is omitted, we replace its conditioning tokens with a learned null token, denoted here $\varnothing$. With task scale w_{task} and context scale w_{ctx}, each Euler step combines three forward passes,

$$\tilde{v} \;=\; w_{\text{task}}\, v_\theta(c^{\text{task}}, c^{\text{ctx}}) \;+\; (w_{\text{ctx}} - w_{\text{task}})\, v_\theta(\varnothing, c^{\text{ctx}}) \;+\; (1 - w_{\text{ctx}})\, v_\theta(\varnothing, \varnothing),$$

equivalent to the unconditional velocity plus a context contribution $w_{\text{ctx}}\big(v_\theta(\varnothing, c^{\text{ctx}}) - v_\theta(\varnothing, \varnothing)\big)$ and a task contribution $w_{\text{task}}\big(v_\theta(c^{\text{task}}, c^{\text{ctx}}) - v_\theta(\varnothing, c^{\text{ctx}})\big)$.

Counterfactual training. Task-fMRI runs contain fixed condition orderings, so the model can overfit to observed condition transitions. We diversify training by splicing discontinuous segments from the same subject (Figure 1b), creating novel condition-to-condition transitions (for a more detailed treatment, see Appendix M). Because splicing introduces an artificial signal discontinuity, we downweight the loss immediately after the splice and inform its restoration trajectory based on the hemodynamic response function. We provide the precise weighting scheme in Appendix L.

4 Experimental Setup

Whereas models typically are assessed on a handful of tasks, both training a model capable of generalizing *across* tasks and performing a robust evaluation requires unique datasets. To this end, we focus on including a large set of varied experimental conditions. This enables evaluation along two axes. First, we verify the generated time series themselves to investigate realism in terms of spectral content, variance, autocorrelation, functional connectivity, and correspondence with real data. Second, we investigate the model's capability to generate data conditioned on tasks unseen during training.

Data. We combine four complementary datasets spanning large subject cohorts and broad task coverage. HCP provides large-scale task-fMRI across seven tasks [Barch et al., 2013], IBC provides dense within-subject coverage across 53 tasks [Ponce et al., 2026], Nakai adds 103 rapidly alternating task conditions for training [Nakai and Nishimoto, 2020], and UK Biobank contributes large-scale

Table 1: Out-of-distribution prediction evaluated by comparing GLM β weights from the real data with those recovered from synthetic data. n are number of conditions or contrasts. Map: spatial map used for conditioning (Direct: Direct's prediction; Cross-run: map from the other phase-encoding acquisition; Oracle: empirical map). We show mean$\pm$std across conditions/contrasts.

		HCP			IBC		
Model	Map	n	Single Cond.	Task Contrasts	n	Single Cond.	Task Contrasts
Zero-resource							
Direct	–	17/13	0.280 ± 0.269	0.262 ± 0.375	191/60	0.245 ± 0.230	0.240 ± 0.243
kNN retrieval	kNN	17/13	0.295 ± 0.308	0.308 ± 0.286	191/60	0.301 ± 0.268	0.317 ± 0.316
Text-FIR	–	17/13	0.482 ± 0.312	0.249 ± 0.226	191/60	0.281 ± 0.331	0.381 ± 0.373
PINC-Flow-Unconditioned	–	17/13	0.435 ± 0.407	0.039 ± 0.219	191/60	0.000 ± 0.197	0.017 ± 0.161
PINC-Flow-f_{txt}	–	17/13	$\mathbf{0.601} \pm 0.313$	$\mathbf{0.346} \pm 0.212$	191/60	$\mathbf{0.418} \pm 0.306$	$\mathbf{0.469} \pm 0.324$
PINC-Flow-f_{spat}	Direct	17/13	0.534 ± 0.251	0.293 ± 0.221	191/60	0.378 ± 0.247	0.407 ± 0.300
Emperical prior							
PINC-Flow-f_{spat}	Cross-run	17/13	0.846 ± 0.131	0.584 ± 0.184	–	–	–
PINC-Flow-f_{spat}	Oracle	17/13	0.883 ± 0.128	0.749 ± 0.158	191/60	0.641 ± 0.211	0.766 ± 0.172
In-distribution							
PINC-Flow-f_{spat}	Oracle	17/13	0.916 ± 0.146	0.889 ± 0.094	191/60	0.700 ± 0.220	0.839 ± 0.133

resting-state data for unconditional training [Miller et al., 2016]. All data are parcellated with the Schaefer-400 atlas, resampled to 1 Hz, low-pass filtered at 0.25 Hz, and parcel-wise z-scored.

We evaluate zero-shot task generation with seven task-held-out folds. Each fold holds out one HCP task and approximately eight IBC tasks, such that every task is held out once; the dynamics model is trained from scratch for each fold. For HCP, we additionally hold out 100 subjects across all tasks, enabling simultaneous evaluation of held-out-task and held-out-subject generalization. Dataset details are provided in Appendix A.

Evaluation. For task-level evaluation, we generate synthetic counterparts of held-out recordings and analyze them with the same GLM used for the real data. Because the model generates windows rather than entire runs in one pass, we tile each held-out recording with non-overlapping generation windows of length T_g, conditioning each window on the task conditioning and leakage-controlled context volumes. The resulting synthetic sequence is then analyzed as a run-level recording. The model receives sensory, instruction, and response captions at every timestep. The evaluation GLM follows standard fMRI practice and jointly fits event regressors: each event regressor is weighted by its duration or fractional occupancy within a volume and convolved with the canonical HRF. We fit identical design matrices to matched real and synthetic samples, yielding contrast maps $\boldsymbol{\beta}_k$ and $\hat{\boldsymbol{\beta}}_k$, which we compare across parcels. For brevity, we use "contrast" to refer both to single event maps and differences between event maps.

To prevent task information from leaking through the context window, context volumes are offset from the generated task segment. When available, we use pre-task periods with fixation conditioning; otherwise, we sample subject-specific context from a different task and mask its task conditioning. Task annotations are provided only for generated timesteps. Real and synthetic GLMs are always fit on matched samples and matched events (further details are given in Appendix O). For f_{txt}, we provide only text embeddings as task conditioning. For f_{spat}, we additionally evaluate three spatial-prior regimes: predicted priors from the Direct model, independent cross-run (estimated on a second run of the held-out task, mimicking a pilot, localizer, or repeated session), and oracle empirical priors (estimated on the evaluated recording), none of which are seen by the dynamics model during training. We refer to this latter setting as f_{spat} unless otherwise specified.

Baseline: To the best of our knowledge, no prior generative model targets out-of-distribution task-fMRI dynamics. We therefore construct multiple baselines. First, we use a strong text-to-contrast baseline (*Direct*; Appendix B) that predicts contrast maps directly from the same language conditioning used by our model. Direct provides both an evaluation anchor for held-out contrast recovery and, when no empirical spatial prior is available, an alternative source of spatial priors for the dynamics model. Unlike our dynamics model, however, Direct predicts static contrast maps rather than full time series, and therefore cannot model temporal dynamics, sample diversity, individual variability, or counterfactual changes to event timing.

Table 2: Spatial prior vs. generated recording (HCP, n=30 pooled conditions and contrasts). q: r(prior, target) ; f: r(simulator output, prior).

		HRF–VAR sim.		PINC-Flow-f_{spat}
Prior source	q	f	r	r
Direct	.274	.995	.273	**.430**
kNN retrieval	.318	.973	.287	–
Cross-run	.654	.991	.644	**.733**
Oracle empirical	.738	.995	.728	**.825**

Table 3: Trajectory similarity for HCP OOD generation. Median [5th, 95th pct.]

	Trajectory r	Paired Δ
Real split-half ref.	.942 [.935, .950]	–
Text-conditioned		
Text-FIR	.354 [.344, .364]	
PINC-Flow-f_{txt}	**.500** [.476, .519]	+.145 [.123, .162]
Oracle prior		
HRF–VAR sim.	.571 [.550, .590]	
PINC-Flow-f_{spat}	**.620** [.602, .643]	+.050 [.027, .074]

Next, we add two families of generative models (Appendix C). HRF-VAR convolves an activation map with the held-out event timings and adds a residual process using PCA/VAR(1). We pass the map predicting by *Direct* (Sim-Direct), a nearest-neighbor map (from the training set; Sim-kNN), or the same priors used for PINC-Flow. Text-FIR is a deterministic linear finite-impulse-response encoder which maps from text embeddings to BOLD. As these latter two baselines output time series, we use the identical evaluation protocol as for PINC-Flow. Finally, we also evaluate our model without task conditioning and fully in-distribution as reference points.

Hyperparameters We use hemispheric encoders to keep the total token sequence length compact despite the multiple token types injected for each timestep. These, as well as the decoders are learned linear projections. We use the Schaefer-400 atlas [Schaefer et al., 2018] similar to recent foundation modeling efforts [Dong et al., 2024, Lane et al., 2026, Gijsen et al., 2026]. Our default parameters are as follows. We use $T_c = 8$, $T_g = 16$, which yields 144 total tokens per window. To train PINC-Flow, we train a DiT backbone with 12 layers and $D = 1024$ for 100K steps (compute use is described in Appendix G). At inference, we perform 64 Euler steps with guidance scales $w_{\text{task}} = w_{\text{ctx}} = 1.5$. We provide further optimization and model parameters in Appendix D.

5 Results

Generated trajectories recover activation patterns of unseen task conditions. We first ask whether trajectories generated under unseen task conditioning recover the activation structure, by comparing $\hat{\beta}_k$ for condition k recovered from real and generated trajectories. Critically, these activation maps are not predicted directly by the model, but rather are recovered from sampled fMRI trajectories. As such, successful contrast recovery requires the model to generate temporally structured task-evoked dynamics rather than merely outputting static activation maps.

The resulting OOD results are shown in Table 1 with visualizations in Figure 3. Text-only generation performs strongly under held-out task conditioning: PINC-Flow-f_{txt} reaches $r = 0.60$ for HCP single conditions and $r = 0.47$ for IBC task contrasts. It thereby performs better than the baselines in all four settings. Empirical but unseen spatial oracle priors further improve performance, with PINC-Flow-f_{spat} reaching $r = 0.88$ and $r = 0.77$, respectively. Cross-run priors are modestly less effective, but remain considerably stronger than f_{txt}. These empirical prior conditions should be interpreted as a spatial-anchoring test rather than a zero-resource prediction benchmark: the supplied maps provide the target spatial hypothesis, but the model must still realize it as temporally structured fMRI trajectories under held-out event, response, and timing compositions. We evaluate in-distribution as well, to provide an approximate upper bound on performance, as it implicitly accounts for contrast reliability and shared evaluation idiosyncrasies (e.g. generating tiled windows).

We note that the large standard deviations result primarily from a diverse task set, rather than noisy model evaluation. For example, the difference between PINC-Flow-f_{txt} and the Direct baseline is significant with $t(280; \text{pooled; paired over tasks}) = 11.70$, $p < 10^{-25}$. The unconditioned PINC-Flow model performs well on HCP single conditions due to the model generating prototypical visual task activations, which maps well to some of its conditions with rapid visual presentation.

Beyond asking whether each generated task map matches the corresponding real task map, we also investigate whether individual regions are recruited appropriately across unseen tasks. For each region, we correlate predicted and real β values across held-out tasks. This evaluates whether the

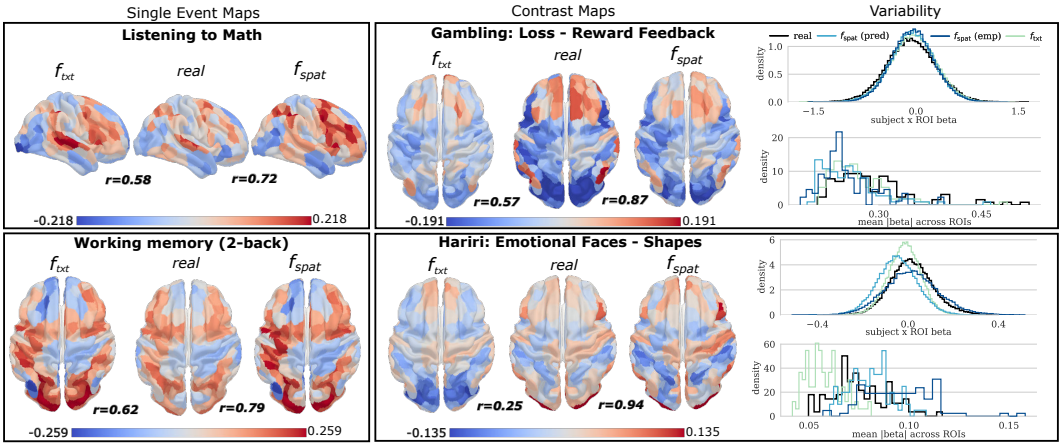


Figure 3: Example β-maps recovered from generated timeseries under unseen task conditioning. We show median cases as well as a failure case in the bottom right, where the contrast map by f_{txt} was not able to capture emotional processing during a perceptual masking task.

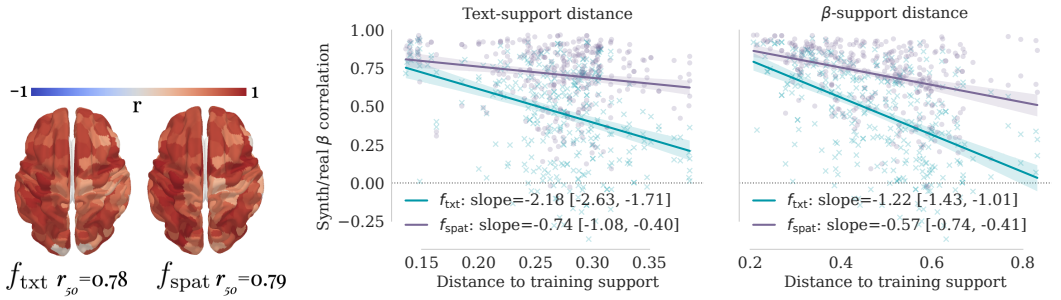


Figure 4: a) Per-region recruitment across tasks. b) The two conditional samplers are compared in their OOD performance as a function of the distance to the training manifold.

model recovers region-specific functions over the OOD task space. We observe high correlations for both f_{txt} (median $r = 0.78$) and f_{spat} (median $r = 0.79$; Figure 4a).

Generated dynamics exceed baselines and their priors. We find that the HRF-VAR simulator effectively passes through the maps it receives, as the correlation between its predictions and the target is consistently capped by the correlation between the supplied prior and the target (Table 2). Meanwhile, PINC-Flow exceeds the prior in each regime (+.16 predicted, +.08 cross-run, +.10 oracle). These results indicate that its advantage is not reducable to retrieval of neighboring tasks or construction of GLM-compatible signals from a supplied prior map.

Temporal trajectories. We compare the similarity of generated trajectories for held-out HCP tasks at the (Yeo-17) network level (Table 3). PINC-Flow-f_{txt} exceeds Text-FIR given identical captions ($\Delta = +.15[.12, .16]$), and the f_{spat} branch exceeds HRF-VAR when both use the oracle prior ($\Delta = +0.05[.03, .07]$). All models scoring considerably lower than the real split-half reference implies that significant progress is still to be made.

Spatial priors stabilize extrapolation away from the training manifold. We next investigate the relationship between the predictability of unseen task conditioning and its distance to the training manifold, pooling single conditions and task contrasts. We compute distance via nearest-neighbor distance to the training manifold in both text and β-space (Figure 4b). As expected, OOD performance degrades as targets move further from the training manifold. However, this degradation is considerably muted for f_{spat} compared to f_{txt}. This supports its intended role: when text conditioning becomes unreliable far from the training manifold, explicit spatial hypotheses can anchor generation while preserving the ability to compose event content, responses, and timing at inference.

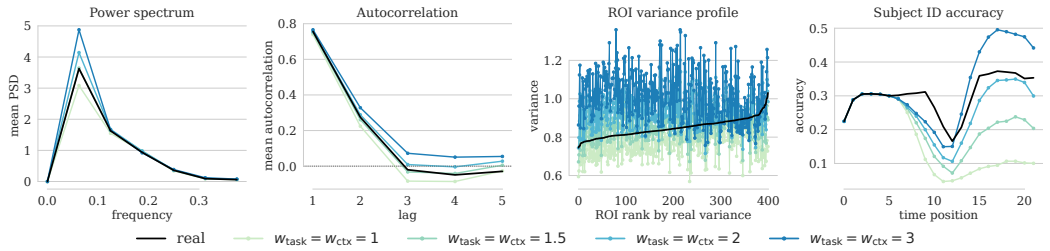


Figure 5: General statistics of the generated data distribution and subject recoverability.

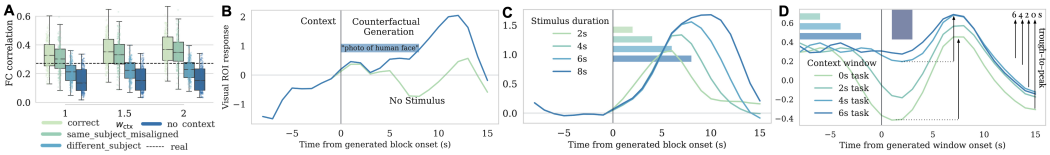


Figure 6: A) Functional connectivity correlations under different context regimes (mean±std). Counterfactual generations are shown for single- (B) and averaged-trajectories (C). D) Longer task presentation in context induces parametric repetition-inhibition effects.

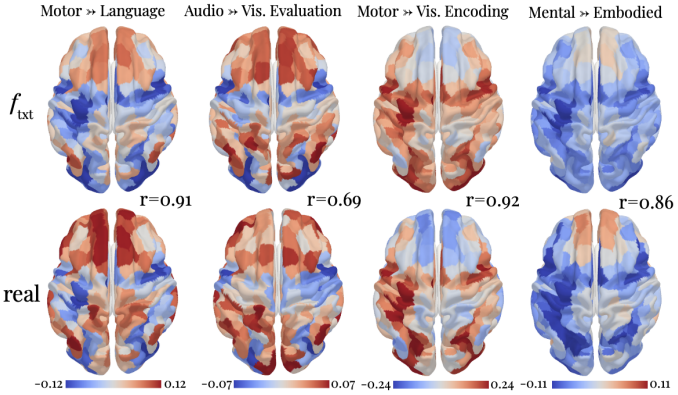
Generated samples exhibit realistic time-series statistics and subject structure. Because the primary evaluation above reduces generated trajectories to GLM contrasts, we next verify that the model also captures broader properties of the time-series distribution under unseen conditioning. We observe that our model produces realistic data in terms of power spectrum, autocorrelation, and variance (Figure 5). Naturally, these statistics depend on the guidance parameters w, which guide generation to emphasize context volumes, task conditioning, or both. We find that the default setting $w_{\text{task}} = w_{\text{ctx}} = 1.5$ produces sensible statistics.

We then ask whether context volumes steer generation toward subject-specific dynamics. A subject-identification classifier trained on real held-out-subject HCP windows recovers subject identity from generated samples, and identity evidence increases with context guidance (Figure 5). Functional connectivity provides a more calibrated test: agreement is highest for the correct context, lower for same-subject misaligned context, lower again for different-subject context, and lowest without context (Figure 6A). This ordering indicates that the model uses context to capture subject-specific structure rather than merely producing generic task dynamics. At high guidance, generated FC can become overly similar to the conditioning context; in Appendix H, we show that stochastic latent sampling reduces this over-conditioning while preserving the expected context ordering.

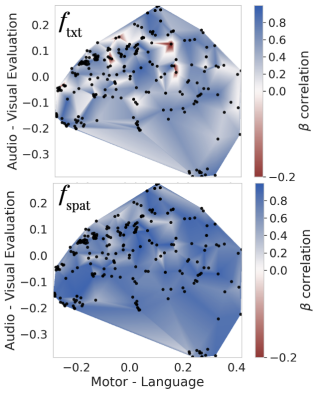
Counterfactual temporal interventions. We study whether the model has learned temporally structured task responses rather than only static spatial associations. We use counterfactual generation to test this, varying both future task conditioning and the content of the context window. The model produces HRF-like responses to counterfactual stimulus presentation, and generated trajectories are sensitive to the recent context history. Interestingly, longer task durations in the context lead to reduced subsequent responses, indicating the model learned dynamics consistent with repetition inhibition (Figure 6; Larsson and Smith [2012]). This supports per-timestep task conditioning as a mechanism for counterfactual interventions on event timing and composition.

We provide an ablation on our splicing training strategy in Appendix F. We find that it particularly helps novel transitions involving in-distribution task conditions. Specifically, splicing removes the tendency to regress towards expected (as per the training distribution) rather than requested conditions.

Generated trajectories preserve semantic structure in task space. Beyond individual contrasts, we test whether generations preserve the organization of task space. We embed each GLM item using its sensory and instruction captions, compute semantic PCA axes, and compare real and synthetic coefficient maps obtained by regressing β values onto these axes. The resulting maps agree across real and generated data, and visualizing the first two semantic axes shows that f_{spat} maintains broader coverage where f_{txt} degrades (Figure 7; Appendix J).



(a) Coefficient maps resulting from regressing β-values onto task space PCA scores.



(b) Task conditions (dots) along their first two PCA axes, colored using the spatial β-map correlation.

Figure 7

Compositional language for generation. Finally, we investigate the effect of dropping language-conditioning components on the synth/real β correlation (Table 4). Each ablation significantly decreases performance, with stronger degradation when multiple conditioning components are removed. This suggests that the model uses the conditioning signal compositionally, while also exhibiting partial redundancy between components. We observe substantial variation in degradation strength across contrasts, which we visualize in Appendix K.

6 Conclusion and Limitations

We introduced PINC-Flow, a per-timestep conditioned diffusion transformer for generating whole-cortex fMRI dynamics under unseen cognitive tasks. Across held-out task conditions, the model recovers task-evoked activation patterns from generated trajectories, preserves region-specific recruitment across task space, and supports counterfactual changes to event timing and composition. This type of model may in the future support (i) screening candidate designs (e.g. amount of task blocks; see Appendix I for an example), (ii) the generation of subject-conditioned synthetic data to validate an analysis pipeline and perform power estimation, and (iii) testing whether a hypothesis yields detectable effects given an experimental design.

Table 4: Synth/real β correlations under hierarchical component dropping of language token $\mathbf{c}^{\text{txt}}$.

$\lvert C\rvert$	Dropped components C	r
0	$\varnothing$ (PINC-Flow-f_{txt})	**.440**
1	r (response)	.410*
	i (instruction)	.397*
	s (sensory)	.395*
2	$i + r$	.350*
	$s + r$	.355*
	$i + s$	.229*
3	$i + s + r$ (*uncond.*)	.023*

$n = 281$. All 12 paired one-sided t-tests over tasks $^{*}p_{\text{Bonf}} < 10^{-4}$.

Some limitations deserve mention. First, our model extrapolates around the task-fMRI manifold covered by the training datasets, rather than to arbitrary cognitive experiments. Although HCP and IBC provide broad task coverage, far-extrapolative text conditions remain difficult and performance degrades with distance from the training support. Further, although we find that f_{spat} improves on a prior map, it's dependent on the quality of the map, which causes a weak predicted prior to underperform conditioning on language alone. Next, our splicing pretraining strategy creates only approximate counterfactuals that retain carryover from the original preceding event. Finally, language conditioning underspecifies fine-grained perceptual content, especially in low-level sensory cortex. Hybrid conditioning with modality-specific encoders is a natural extension, while our results indicate that the current language pathway remains well matched to whole-cortex cognitive dynamics.

7 Acknowledgements and Disclosures

This research was funded by Gemeinnützigen Hertie-Stiftung and the Deutsche Forschungsgemeinschaft (DFG) through FOR 5187 (project number 442075332). Additional support was provided by the Machine Excellence Cluster and DFG through the Germany's Excellence Strategy (EXC 2064 - project number 390727645) and the following projects: CRC 1404 (project number 414984028), TRR 265 (project number 402170461), and RU 5363 (project number 459422098).

We thank Moritz Seiler and Nicolas Münster for their valuable comments on an early version of this manuscript. Finally, we are grateful to all dataset providers for access (UKB application number: 25163).

The authors declare no competing interests.

References

Deanna M Barch, Gregory C Burgess, Michael P Harms, Steven E Petersen, Bradley L Schlaggar, Maurizio Corbetta, Matthew F Glasser, Sandra Curtiss, Sachin Dixit, Cindy Feldt, et al. Function in the human connectome: task-fmri and individual differences in behavior. *Neuroimage*, 80: 169–189, 2013.

Victoria Bosch, Daniel Anthes, Adrien Doerig, Sushrut Thorat, Peter König, and Tim Christian Kietzmann. Brain-language fusion enables interactive neural readout and in-silico experimentation. *arXiv preprint arXiv:2509.23941*, 2025.

Zijian Dong, Ruilin Li, Yilei Wu, Thuan T Nguyen, Joanna S Chong, Fang Ji, Nathanael R Tong, Christopher L Chen, and Juan H Zhou. Brain-jepa: Brain dynamics foundation model with gradient positioning and spatiotemporal masking. *Advances in Neural Information Processing Systems*, 37: 86048–86073, 2024.

Stéphane d'Ascoli, Jérémy Rapin, Yohann Benchetrit, Teon Brookes, Katelyn Begany, Joséphine Raugel, Hubert Banville, and Jean-Rémi King. A foundation model of vision, audition, and language for in-silico neuroscience. *arXiv preprint arXiv:2605.04326*, 2026.

Karl J Friston, Paul Fletcher, Oliver Josephs, ANDREW Holmes, Michael D Rugg, and Robert Turner. Event-related fmri: characterizing differential responses. *Neuroimage*, 7(1):30–40, 1998.

Shachar Gal, Yael Coldham, Niv Tik, Michal Bernstein-Eliav, and Ido Tavor. Act natural: Functional connectivity from naturalistic stimuli fmri outperforms resting-state in predicting brain activity. *NeuroImage*, 258:119359, 2022.

Yunlong Gao, Jinbo Yang, Li Xiao, Haiye Huo, Yang Ji, Hao Wang, Aiying Zhang, and Yu-Ping Wang. Braincast: A spatio-temporal forecasting model for whole-brain fmri time series prediction. *arXiv preprint arXiv:2603.13361*, 2026. URL `https://arxiv.org/abs/2603.13361`.

Sam Gijsen and Kerstin Ritter. EEG-language pretraining for highly label-efficient clinical phenotyping. In *Proceedings of the 42nd International Conference on Machine Learning*. PMLR, 2025. URL `https://proceedings.mlr.press/v267/gijsen25a.html`.

Sam Gijsen, Marc-Andre Schulz, and Kerstin Ritter. Brain-semantoks: Learning semantic tokens of brain dynamics with a self-distilled foundation model. In *International Conference on Learning Representations*, 2026. arXiv:2512.11582.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Emiel Hoogeboom, Vıctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pages 8867–8887. PMLR, 2022.

Yufan Hu, Wuyang Li, and Yixuan Yuan. Synthesizing realistic fmri: a physiological dynamics-driven hierarchical diffusion model for efficient fmri acquisition. In *The Thirteenth International Conference on Learning Representations*, 2025.

Ilia Igashov, Hannes Stärk, Clément Vignac, Arne Schneuing, Victor Garcia Satorras, Pascal Frossard, Max Welling, Michael Bronstein, and Bruno Correia. Equivariant 3d-conditional diffusion model for molecular linker design. *Nature Machine Intelligence*, 6(4):417–427, 2024.

John B Ingraham, Max Baranov, Zak Costello, Karl W Barber, Wujie Wang, Ahmed Ismail, Vincent Frappier, Dana M Lord, Christopher Ng-Thow-Hing, Erik R Van Vlack, et al. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, 2023.

Connor Lane, Mihir Tripathy, Leema Krishna Murali, Ratna Sagari Grandhi, Shamus Sim Zi Yang, Sam Gijsen, Debojyoti Das, Manish Ram, Utkarsh Kumar Singh, Cesar Kadir Torrico Villanueva, Yuxiang Wei, Will Beddow, Gianfranco Cortés, Suin Cho, Daniel Z. Kaplan, Benjamin Warner, Tanishq Mathew Abraham, and Paul S. Scotti. Scaling vision transformers for functional mri with flat maps. *arXiv preprint arXiv:2510.13768*, 2026. doi: 10.48550/arXiv.2510.13768. Accepted to ICML 2026.

Jonas Larsson and Andrew T Smith. fmri repetition suppression: neuronal adaptation or stimulus expectation? *Cerebral cortex*, 22(3):567–576, 2012.

Seul Lee, Jaehyeong Jo, and Sung Ju Hwang. Exploring chemical space with score-based out-of-distribution generation. In *International Conference on Machine Learning*, pages 18872–18892. PMLR, 2023.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. *arXiv preprint arXiv:2301.12503*, 2023.

Nikos K Logothetis, Jon Pauls, Mark Augath, Torsten Trinath, and Axel Oeltermann. Neurophysiological investigation of the basis of the fmri signal. *nature*, 412(6843):150–157, 2001.

Mohammad Lotfollahi, F Alexander Wolf, and Fabian J Theis. scgen predicts single-cell perturbation responses. *Nature methods*, 16(8):715–721, 2019.

Mohammad Lotfollahi, Anna Klimovskaia Susmelj, Carlo De Donno, Leon Hetzel, Yuge Ji, Ignacio L Ibarra, Sanjay R Srivatsan, Mohsen Naghipourfar, Riza M Daza, Beth Martin, et al. Predicting cellular responses to complex perturbations in high-throughput screens. *Molecular systems biology*, 19(6), 2023.

Andrew Luo, Maggie Henderson, Leila Wehbe, and Michael Tarr. Brain diffusion for visual exploration: Cortical discovery using large scale generative models. *Advances in Neural Information Processing Systems*, 36:75740–75781, 2023.

Karla L Miller, Fidel Alfaro-Almagro, Neal K Bangerter, David L Thomas, Essa Yacoub, Junqian Xu, Andreas J Bartsch, Saad Jbabdi, Stamatios N Sotiropoulos, Jesper LR Andersson, et al. Multimodal population brain imaging in the uk biobank prospective epidemiological study. *Nature neuroscience*, 19(11):1523–1536, 2016.

Tomoya Nakai and Shinji Nishimoto. Quantitative models reveal the organization of diverse cognitive functions in the brain. *Nature communications*, 11(1):1142, 2020.

Thomas Naselaris, Kendrick N Kay, Shinji Nishimoto, and Jack L Gallant. Encoding and decoding in fmri. *Neuroimage*, 56(2):400–410, 2011.

Furkan Ozcelik and Rufin VanRullen. Natural scene reconstruction from fmri signals using generative latent diffusion. *Scientific Reports*, 13(1):15666, 2023.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

Ana Fernanda Ponce, Himanshu Aggarwal, Swetha Shankar, Juan Jesús Torre, Ana Luísa Pinho, Alexis Thual, Chantal Ginisty, Yann Lecomte, Valérie Berland, Lucile Beriot, et al. Individual brain charting: fifth release of high-resolution fmri data for cognitive mapping. *Scientific data*, 2026.

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

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Alexander Schaefer, Ru Kong, Evan M Gordon, Timothy O Laumann, Xi-Nian Zuo, Avram J Holmes, Simon B Eickhoff, and BT Thomas Yeo. Local-global parcellation of the human cerebral cortex from intrinsic functional connectivity mri. *Cerebral cortex*, 28(9):3095–3114, 2018.

Arne Schneuing, Charles Harris, Yuanqi Du, Kieran Didi, Arian Jamasb, Ilia Igashov, Weitao Du, Carla Gomes, Tom L Blundell, Pietro Lio, et al. Structure-based drug design with equivariant diffusion models. *Nature Computational Science*, 4(12):899–909, 2024.

Paul Scotti, Atmadeep Banerjee, Jimmie Goode, Stepan Shabalin, Alex Nguyen, Aidan Dempster, Nathalie Verlinde, Elad Yundler, David Weisberg, Kenneth Norman, et al. Reconstructing the mind's eye: fmri-to-image with contrastive learning and diffusion priors. *Advances in Neural Information Processing Systems*, 36:24705–24728, 2023.

Jungwoo Seo, David Keetae Park, Shinjae Yoo, and Jiook Cha. Scalable diffusion transformer for conditional 4d fmri synthesis. *arXiv preprint arXiv:2511.22870*, 2025.

Emin Serin, Kerstin Ritter, Gunter Schumann, Tobias Banaschewski, Andre Marquand, and Henrik Walter. Generating synthetic task-based brain fingerprints for population neuroscience using deep learning. *Communications Biology*, 8(1):1572, 2025.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. pmlr, 2015.

Denes Szucs and John PA Ioannidis. Sample size evolution in neuroimaging research: An evaluation of highly-cited studies (1990–2012) and of latest practices (2017–2018) in high-impact journals. *NeuroImage*, 221:117164, 2020.

Ido Tavor, O Parker Jones, Rogier B Mars, Stephen M Smith, Timothy E Behrens, and Saad Jbabdi. Task-free mri predicts individual differences in brain activity during task performance. *Science*, 352(6282):216–220, 2016.

Hwa Hui Tew, Junn Yong Loo, Yee-Fan Tan, Xinyu Tang, Hernando Ombao, Fuad Noman, Raphael C-W Phan, and Chee-Ming Ting. T2i-diff: fmri signal generation via time-frequency image transform and classifier-free denoising diffusion models. *arXiv preprint arXiv:2509.20822*, 2025. URL `https://arxiv.org/abs/2509.20822`.

Julius Vetter, Jakob H Macke, and Richard Gao. Generating realistic neurophysiological time series with denoising diffusion probabilistic models. *Patterns*, 5(9), 2024.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

Yuxiang Wei, Yanteng Zhang, Xi Xiao, Chengxuan Qian, Tianyang Wang, and Vince D Calhoun. fmri-lm: Towards a universal foundation model for language-aligned fmri understanding. *arXiv preprint arXiv:2511.21760*, 2025.

Junfeng Xia, Wenhao Ye, Xuanye Pan, Xinke Shen, Mo Wang, and Quanying Liu. Brain-dit: A universal multi-state fmri foundation model with metadata-conditioned pretraining. *arXiv preprint arXiv:2604.12683*, 2026.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. C-pack: Packaged resources to advance general chinese embedding, 2023.

BT Thomas Yeo, Fenna M Krienen, Jorge Sepulcre, Mert R Sabuncu, Danial Lashkari, Marisa Hollinshead, Joshua L Roffman, Jordan W Smoller, Lilla Zöllei, Jonathan R Polimeni, et al. The organization of the human cerebral cortex estimated by intrinsic functional connectivity. *Journal of neurophysiology*, 2011.

Rongquan Zhai, Yechen Hu, Liping Zheng, Shitong Xiang, Chao Xie, Lei Peng, Tobias Banaschewski, Gareth J Barker, Arun LW Bokde, Rüdiger Brühl, et al. Brain diffusion transformer for personalized neuroscience and psychiatry. *bioRxiv preprint*, 2025. URL `https://www.biorxiv.org/content/10.1101/2025.04.12.648506v1`.

A Dataset Details

We combine four datasets that differ in subject count, task diversity, and recording structure. The Human Connectome Project (HCP; Barch et al. [2013]) provides approximately $1{,}100$ subjects across seven tasks, supporting held-out-subject evaluation. The Individual Brain Charting dataset (IBC; [Ponce et al., 2026]) includes 12 subjects who each perform 53 tasks, providing broad task-space coverage with at least one dedicated recording run per task and subject. The Nakai dataset [Nakai and Nishimoto, 2020] comprises 6 subjects across 103 task conditions. Because Nakai task conditions alternate rapidly within each run across 18 runs per subject, we use it only for training to avoid biased task-level evaluation through temporal leakage. UK Biobank (UKB; Miller et al. [2016]) contributes approximately $39{,}000$ subjects of resting-state data, used unconditionally during training to improve modeling of intrinsic brain activity.

We match the sampling rate to 1 s across datasets. Since the slowest original repetition time is 2 s, we apply a 0.25 Hz low-pass filter before resampling and z-score each parcel. Overall, the combined dataset contains 3.6M task-fMRI timepoints and 13.9M resting-state timepoints. Due to dataset imbalance, batches are sampled with non-uniform dataset weights to preserve gradient pressure on task data:

$$w_{\text{HCP}} = 1.0, \quad w_{\text{IBC}} = 0.85, \quad w_{\text{Nakai}} = 0.15, \quad w_{\text{UKB}} = 0.2.$$

Access. Data access is governed by the original providers of each dataset and can be requested through the procedures described in the respective dataset citations. All datasets require a data-use or data-sharing agreement, and all have been widely used in prior neuroimaging and machine learning research.

B Direct Text-to-Contrast Baseline

The Direct baseline predicts static contrast maps from the same language-derived task annotations used by the dynamics model. It decomposes each timestep's conditioning into an event component and, when applicable, a response component. Event components $\hat{\boldsymbol{\beta}}^{\text{event}}$ are predicted from the concatenated sensory and instruction embeddings using ridge regression. Response components $\hat{\boldsymbol{\beta}}^{\text{resp}}$ are predicted by nearest-neighbor lookup over the training fold, reflecting the small and discrete response space. When both event and response components are present, the predicted map is their sum; otherwise, the available component is used directly.

Hyperparameters are selected by fold-internal cross-validation, with no held-out task leakage. We search ridge penalties

$$\alpha \in \{1, 10, 100, 10^3, 10^4, 10^5\}$$

and response nearest-neighbor values

$$k \in \{1, 5, 10, 20\}.$$

Model selection uses Pearson correlation between predicted and held-out training-fold contrast maps. We found that strong ridge regularization is typically selected, typically $\alpha = 10^4$, which improves spatial pattern recovery but shrinks prediction magnitudes. To restore realistic amplitudes without using held-out tasks, we rescale predictions within each Yeo-17 network to match the training-fold contrast norms. These networks by Yeo et al. [2011] are widely-used in the neuroscience literature. The resulting Direct maps are used both as a baseline in contrast-recovery analyses and as optional predicted spatial priors for the dynamics model.

C Generative Baselines

HRF-VAR. Given the held-out task schedule $s_k(t)$ and map $\hat{\boldsymbol{\beta}}_k$,

$$\hat{\mathbf{x}}_t = \sum_k \hat{\boldsymbol{\beta}}_k \, (\mathrm{hrf} * s_k)(t) + \mathbf{U}\boldsymbol{\eta}_t, \qquad \boldsymbol{\eta}_t = \mathbf{A}\boldsymbol{\eta}_{t-1} + \boldsymbol{\xi}_t, \;\; \boldsymbol{\xi}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}), \tag{1}$$

where $\mathbf{U}$ denote PCA loadings and $(\mathbf{A}, \boldsymbol{\Sigma})$ are the VAR(1) parameters fit on the GLM residuals. Number of PCA components was selected via cross-validation from $\{10, 20, 30, 40, 50\}$, with a post-hoc analysis showing that the maximum was selected almost always. On average, fits were modestly better going from 40 to 50 components, indicating that more components might have yielded slightly better results.

Text-FIR. We use the same caption embedding $\mathbf{e_t}$ as used for PINC-Flow and fit using ridge regression:

$$\hat{\mathbf{x}}_t = \sum_{\ell=0}^{L-1} \mathbf{W}_\ell \, \mathbf{e}_{t-\ell} + \mathbf{b}, \tag{2}$$

The regularization strength was selected from $\{10^2, 10^3, 10^4, 10^5, 10^6\}$, with 10^5 being selected for every fold.

D Additional Model and Optimization Hyperparameters

Table 5: Summary of model architecture, dropout probabilities, and optimization hyperparameters.

Hyperparameter	**Value**
Architecture & Dimensions	
Embedding dimension	1024
Number of heads	16
Number of layers	12
MLP ratio	4.0
Dropout	0.05
Context frames	8
Generation frames	16
Max context length	24
Dropout Probabilities (p_{drop})	
Spatial prior	0.5
Context volumes	0.2
Condition context[a]	0.15
Condition future[b]	0.1
Instruction embedding	0.1
Sensory embedding	0.1
Response embedding	0.1
Training & Optimization	
Batch size	256
Number of epochs	1000 (100 steps/epoch)
Mixed precision (`use_amp`)	True
Gradient accumulation steps	1
Optimizer	AdamW
Learning rate	2.4×10^{-4}
Weight decay	0.05
AdamW betas	$(0.9, 0.999)$
Learning rate schedule	Cosine decay with warmup[c]
Warmup epochs	10
Minimum LR ratio	0.3
Cosine epochs	500
Text Embedding Model	
Model	BAAI/bge-large-en-v1.5 [Xiao et al., 2023]
Dimensionality	1024

[a] All task conditioning for context volumes.
[b] All task conditioning for generation volumes.
[c] After warmup, cosine decay over `cosine_epochs`, then constant at `min_lr_ratio` $\times$ `lr`.

E Flow matching parameterization

In the flow matching paradigm, models are commonly trained to predict the velocity. With the linear path $z_\tau = (1-\tau)z_0 + \tau z_1$, the velocity is $v = (z_1 - z_\tau)/(1-\tau)$. We here use an alternative to this "v-pred" strategy, which is to predict the endpoint ("x-pred") using the loss $|\hat{z}_1 - z_1|^2$. We note that, via substitution:

$$|\hat{z}_1 - z_1|^2 = (1-\tau)^2 |\hat{v} - v|^2$$

Therefore, training with "x-pred" is the same as training on the velocity loss with weight $w(\tau) = (1-\tau)^2$, yielding the same optimum but differing in the contribution of each point along the path. A priori, our motivation was that the endpoint target is a clean data sample at a fixed scale. At high τ, z_1 is almost entirely determined by z_τ and almost not at all by z_0. As such, the velocity target $z_1 - z_0$ contains irreducible conditional variance while the endpoint target does not. We run

a post-hoc ablation, but find there to be a minimal and likely neglible difference between the two parameterizations. Here we show v-pred minus x-pred:

Metric	Item-weighted Δ	Task-macro Δ
Contrasts	$-.020$	~ 0.000
Single conditions	$-.041$	$-.029$

F Splicing ablation

We retrained models without splicing. While OOD performance is unchanged, we find that transitions involving in-distribution tasks are degraded. Specifically, we condition models using contexts containing task block A while requesting C, whereas B was the observed successor during training. We find that splicing modestly improves recovery of C ($r = .74 \rightarrow .81$) and removes reversion to the context-implied B ($r = .17 \rightarrow -.02$) (Table 6).

Table 6: Transition-splicing ablation (PINC-Flow-f_{txt}). We test in-distribution (ID), where the context seen during training implies block B will follow, but block C is requested instead (paired leave-one-subject-out CI). For OOD, we show map recovery over all seven held-out-task folds (mean$\pm$std over items; CI: bootstrap over 52 tasks). Note that results differ minimally from the main results: for computational efficiency we only evaluate on one (rather than all six) captions.

	No splice	Splice	Δ [95% CI]
ID: counterfactual (MOTOR)			
r w/ requested C $\uparrow$	.742	**.811**	$+.069$ [.052, .085]
r w/ obsolete B $\downarrow$	.166	$\mathbf{-.025}$	$-.191$ [$-.212$, $-.171$]
OOD: map recovery			
r (all items)	$.433 \pm .308$	$.423 \pm .315$	$-.010$ [$-.034$, .012]

G Compute Usage

Training our model for 100K steps takes 1.5 days on a single L40 GPU (48GB). Across seven folds, this results in an aggregate 10.5 days.

While evaluation costs themselves are negligible, they require model inference for synthetic data generation. This takes approximately three hours per fold on a 20GB GPU. The main results compare five models across 7 folds, yielding an approximate 105 hour compute usage. We ablate two models under 4 guidance scales and one model under 7 dropout settings, yielding another 315 hours of 20GB GPU usage.

In total, the experiments reported in the paper therefore use about 672 GPU hours. We estimate preliminary work to have consumed twice this amount at approximately 1400 GPU hours.

H Stochastic Sampling

All main task-generation results use deterministic Euler sampling. We additionally evaluate a stochastic sampler to test whether context-related over-conditioning can be reduced without changing the trained model. At each Euler step except the final step, we add temporally correlated AR(1) Gaussian noise in latent space with autocorrelation $\rho = 0.7$ and scale $\varepsilon = 0.6$, scaled by $\sqrt{\Delta\tau}(1-\tau)$ so that stochasticity decreases as the sampler approaches $\tau = 1$.

Deterministic sampling produces realistic task contrasts and basic time-series statistics, but generated FC can be more similar to the conditioning context than expected from real run-to-run FC. Stochastic sampling improves calibration, reducing generated-context FC correlations toward the real reference distribution while preserving the expected monotonic effect of context guidance.

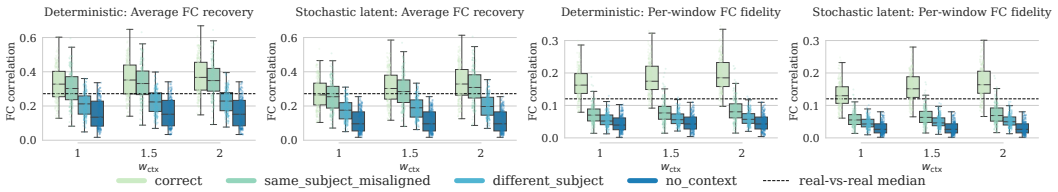


Figure 8: Functional connectivity in generated trajectories can be modulated via guidance and is better calibrated using stochastic sampling. Mean±std across subjects (left) and windows (right).

I Design reliability screening example

As an example future use-case for conditional generative models, we investigate the model for design screening (Figure 9). Concretely, for the HCP language task, we generate between one and four task blocks and compute the "story > math" map. On the left, we plot the split-half reliability across the number of task blocks. We find that our DiT model (using an independent cross-run prior) does well and captures a stable group map after 2-3 blocks. In the middle, we measure within-source sampling error, here PINC-Flow again reproduces the shape of the noise-vs-blocks curve, although it underestimates the noise compared to real data. This would lead to an underestimating of how many blocks are needed. Finally, on the right, the error with respect to a real map shows that our model has a bias-based error for this particular OOD task, as error does not significantly reduce with more data. These post-hoc analyses were performed with default sampling parameters and we leave attempts to improve this type of calibration via training and inference tuning for future work.

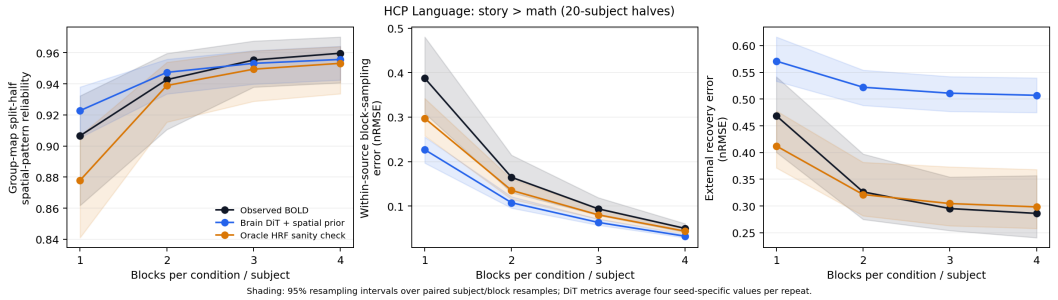


Figure 9: Using the HCP Language task, we investigate the design screening capability of PINC-Flow.

J Task Space PCA

Following PCA of the concatenated sensory and instruction text embeddings, we investigated the resulting low-dimensional semantic axes. To interpret these axes, we projected all experimental events onto the first five PCs. We then examined the items (comprising single conditions and task contrasts) with the high positive and negative dot product scores for each PC. Top and bottom four scores are presented in Table 7. This analysis revealed clear semantic distinctions for four of the five axes. For instance, PC 1 discriminates between tasks involving language processing (e.g., “auditory theory of mind,” “audio sentence,” “listen to the story,” “general auditory,” “audio story about false beliefs”) and motor execution (e.g., “move right/left hand,” “move right/left foot”). PC 2 differentiates visual evaluation tasks (e.g., gamble presentation, preferences for faces/paintings/houses/food) from auditory stimulation (e.g., listening to different sounds). PC 3 separates visual encoding (e.g., dots/letters stimuli) from motor tasks. PC 5 contrasts embodied processes (e.g., moving tongue/hand, face preference, emotion recognition) with mental simulation tasks (e.g., mental time travel west-east/north-south). However, we did not find a clearly identifiable semantic boundary in PC 4, therefore it was excluded from the visualization.

Table 7: Semantic PCA axes 1-5 - conditions and contrasts text representations projected onto the respective PC axes and sorted by the resulted dot product (score). We present the first and last 4 values. Item kinds are abbreviated as s. (single condition) and c. (contrast).

axis	rank	score	dataset	task	kind	item name
1	1	2.355	IBC	MathLanguage	s.	events:theory_of_mind_auditory
	2	2.332	IBC	ArchiStandard	s.	events:audio_sentence
	3	2.293	HCP	LANGUAGE	s.	story_listen
	4	2.276	IBC	MathLanguage	s.	events:general_auditory
	⋮	⋮	⋮	⋮	⋮	⋮
	-4	-1.588	HCP	MOTOR	c.	move_rf_gt_fixation
	-3	-1.588	HCP	MOTOR	s.	move_rf
	-2	-1.612	HCP	MOTOR	c.	move_rh_gt_fixation
	-1	-1.612	HCP	MOTOR	s.	move_rh
2	1	1.905	IBC	NARPS	s.	events:narps_gamble_gain_hi_loss_hi
	2	1.902	IBC	NARPS	s.	events:narps_gamble_gain_lo_loss_hi
	3	1.901	IBC	NARPS	c.	stim_gt_fixation
	4	1.900	IBC	NARPS	s.	events:narps_gamble_gain_hi_loss_lo
	⋮	⋮	⋮	⋮	⋮	⋮
	-4	-2.647	IBC	Audi	s.	events:french_speech
	-3	-2.672	IBC	Audi	s.	events:alphabet_audio
	-2	-2.678	IBC	Audi	s.	events:coughing_sounds
	-1	-2.687	IBC	Audi	c.	all_audio_gt_silence
3	1	1.918	IBC	MVIS	s.	events:six_dots_array
	2	1.851	IBC	MVIS	s.	events:four_dots_array
	3	1.846	IBC	MVEB	s.	events:four_letters_same_string
	4	1.811	IBC	MVEB	c.	string_encoding_gt_fixation
	⋮	⋮	⋮	⋮	⋮	⋮
	-4	-2.176	HCP	MOTOR	s.	move_rf
	-3	-2.205	HCP	MOTOR	s.	move_rh
	-2	-2.205	HCP	MOTOR	c.	move_rh_gt_fixation
	-1	-2.230	IBC	MTTWE	s.	events:westside_event_audio
4	1	1.946	IBC	MVEB	s.	events:two_letters_different_string
	2	1.934	IBC	MVEB	c.	string_encoding_gt_fixation
	3	1.924	IBC	MVEB	s.	events:two_letters_same_string
	4	1.900	IBC	MVEB	s.	events:four_letters_different_string
	⋮	⋮	⋮	⋮	⋮	⋮
	-4	-2.717	IBC	BiologicalMotion2	s.	events:modified_inverted
	-3	-2.729	IBC	BiologicalMotion2	s.	events:natural_upright
	-2	-2.768	IBC	BiologicalMotion2	c.	biological_motion_gt_fixation
	-1	-2.799	IBC	BiologicalMotion2	s.	events:natural_inverted
5	1	1.894	IBC	Moto	s.	events:tongue_left
	2	1.884	IBC	Moto	s.	events:hand_left
	3	1.848	IBC	Moto	s.	events:tongue_right
	4	1.815	IBC	Moto	c.	saccade_gt_other_movements
	⋮	⋮	⋮	⋮	⋮	⋮
	-4	-3.455	IBC	MTTNS	s.	events:southside_close_event
	-3	-3.478	IBC	MTTNS	c.	all_events_gt_fixation
	-2	-3.478	IBC	MTTNS	c.	space_event_gt_time_event
	-1	-3.483	IBC	MTTNS	s.	events:northside_close_event

K Language-Token component dropout - extra results

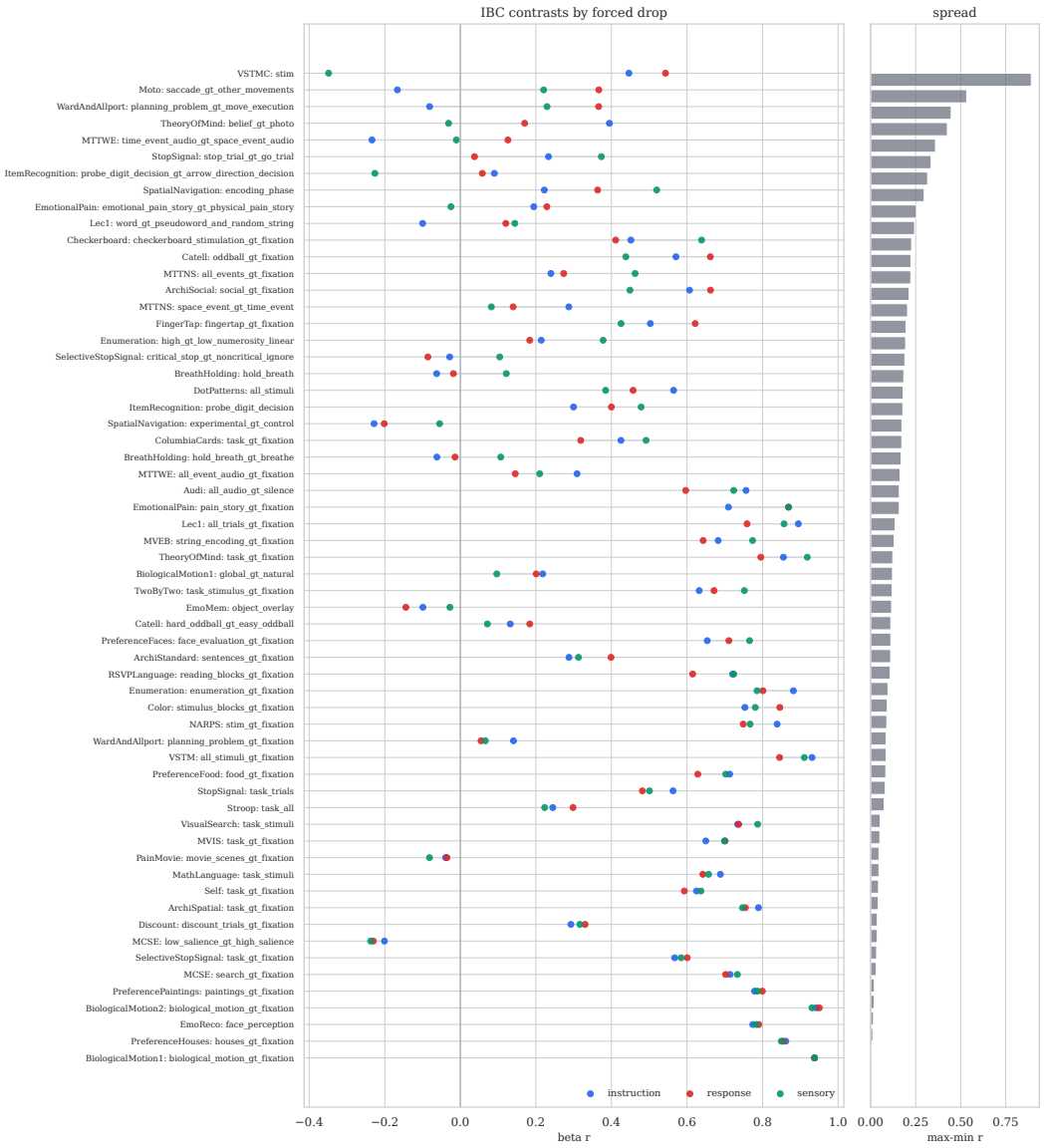


Figure 10: We show the effects on synth/real β correlations for individual IBC contrasts when dropping out a single language token components during OOD evaluation.

L Counterfactual Training details

fMRI runs use fixed condition orderings, so the model repeatedly observes the same condition-to-condition transitions and tends to overfit to them, which harms compositional generalization. To diversify transitions, with probability p_{cf} we splice two discontinuous segments from the same subject, creating a novel condition transition with a temporal discontinuity at splice point t_s.

We therefore weight the per-timestep loss for $t > t_s$ by

$$\tilde{w}_t = \begin{cases} \dfrac{hrf(t - t_s)}{hrf(\tau_{\text{peak}})}, & t_s \le t \le t_s + \tau_{\text{peak}}, \\ 1, & t_s + \tau_{\text{peak}} < t \le T_g, \end{cases} \qquad w_t = \frac{T_g}{\sum_{u=1}^{T_g} \tilde{w}_u} \tilde{w}_t.$$

Here hrf is the canonical double-gamma HRF and $\tau_{\text{peak}} \approx 5\,\text{s}$ is the location of its peak. The unnormalized weight $\tilde{w}_t$ is zero at the splice point, where the model cannot reconstruct the discontinuity, and recovers smoothly over the HRF rise time. We rescale to w_t so that $\sum_{t=1}^{T_g} w_t = T_g$, preserving the average loss scale over the generation window across spliced and unspliced samples.

M Counterfactual interpretation and splice approximation

We use the term counterfactual to mean the model-implied response under a task composition, transition, or timing not observed during training. For manipulations covered by the training data, the generated distribution can be seen as an interventional distribution under consistency, no interference, and positivity. For held-out tasks, the model must instead extrapolate from components seen during training. Finally, conditional flows do not identify how an individual's response for a given task pairs with their response for another task. As such, we claim we perform counterfactual simulation, rather than identified individual potential outcomes.

Interventional reading. Let c denote a task specification controlled by an experimenter. Under consitency (the observed response equals the potential response under the realized c), no interference (one subject's response is unaffected by another's assignment), and positivity (each c of interest has support in training), $p_1(z_1|c^{\text{txt}}, c^{\text{ctx}})$ coincides with the distribution $p(z_1|\text{do}(c), c^{\text{ctx}})$. For held-out tasks, positivity fails as generation extrapolates over components.

Splice decomposition. Let $\mathbf{x}_B(t|A)$ denote the response to event B following the event A, $\mathbf{x}_B(t|A) = \mathbf{r}_B(t|A) + \mathbf{c}_A(t) + \varepsilon_t$, where $\mathbf{r}_B$ is the event-related respones to B, $\mathbf{c}_A$ the residual activity bleeding from A, and ε_t noise. Splicing then resuses a segment observed after A in place of the unobserved response after C:

$$\mathbf{x}_B(t \mid A) - \mathbf{x}_B(t \mid C) = \underbrace{\mathbf{r}_B(t \mid A) - \mathbf{r}_B(t \mid C)}_{\text{history-dependent modulation}} + \underbrace{\mathbf{c}_A(t) - \mathbf{c}_C(t)}_{\text{predecessor carryover}} + (\boldsymbol{\varepsilon}_t - \boldsymbol{\varepsilon}'_t).$$

At the β-map level, the splice is an approxiamte counterfactual target when B's spatial response is locally stable across precedessors. For the timeseries level however, carryover decays approximately over one HRF, which motivates the HRF-based loss weighting.

Non-identifiability of pairings. Conditional flows match the task-specific marginals $p(z_1|c)$. Different latent parameterizations can reproduce the same marginals while yielding different pairings of an individual's responses cross-task. The generated cross-task pairs are therefore only 'model-implied', rather than identified potential outcomes.

N Extended related work

Encoding or generative models. TRIBEv2 [d'Ascoli et al., 2026] is a stimulus-encoding model, as it maps audio, video, and text streams to a deterministic, group-level BOLD trajectory. Meanwhile, PINC-Flow conditions on a factorized, textual experimental description, subject context, and an optional spatial prior to sample a distribution over recordings. We incude motor responses and task instructions to also model active rather than only perceptual paradigms. Conversely, the raw-stimulus encoding used by TRIBEv2 is better situated to preserve perceptual details missed by language descriptions. BrainDiVE [Luo et al., 2023] and BeNeDiff [Lee et al., 2023] are methods using diffusion in the opposite direction. As they generate images or behaviour videos associated with brain factors, they ask what neural factors represents, while we ask what neural dynamics follow from an experimental condition. We provide a visual comparison across these methods in Figure 11.

	neural activity → content		experiment → neural activity	
	BrainDiVE	**BeNeDiff**	**TRIBE v2**	**PINC-Flow** (ours)
Generated output	images	behavior video	fMRI	fMRI
Conditioning signal	ROI target	neural latents	raw stimuli	captions
Recording modality	human fMRI	mouse Ca^{2+}	human fMRI	human fMRI
Generates neural time series	○	○	●	●
Samples a distribution	●	●	○	●
Raw stimulus input	–	–	●	○
Instructions & motor responses as inputs	–	–	○	●
Per-timestep control of event timing	–	–	●	●
Steerable by a spatial / neural hypothesis	●	●	○	●
Individualized output for new subjects w/o retraining	○	–	◐	●
Novel stimuli at inference	–	–	●	◐
Held-out task paradigms	–	–	●	●
Subcortical targets	–	–	●	○

● native ◐ partial / indirect ○ not supported – not applicable

Figure 11: Methods comparison.

O Synthetic Run Construction and GLM Evaluation

Windowed generation. The dynamics model generates fixed-length windows rather than full runs in a single forward process. For evaluation, we therefore construct synthetic counterparts of held-out task recordings by repeatedly generating non-overlapping windows of length T_g. Each generated window receives the task annotations for the corresponding target timesteps and leakage-controlled context volumes. The generated windows are then concatenated into a synthetic run-like sequence and analyzed with the same GLM machinery as the corresponding real data.

For HCP, generation windows begin at the start of each task block. This covers nearly all task-relevant volumes, although short inter-block or trailing periods may not be represented. HCP also includes pre-task and fixation periods, allowing context volumes to be drawn from periods whose task conditioning can be truthfully specified as fixation. For IBC, which does not provide the same block structure, we generate non-overlapping windows along the target recording. Because IBC provides little pre-task signal, context is drawn from a different task recording from the same subject; task-conditioning tokens are masked for these context timesteps to prevent leakage. Across datasets, task conditioning is supplied only for generated volumes.

Matched real and synthetic GLMs. Although the generative model receives sensory, instruction, and response captions at every timestep, evaluation follows standard task-fMRI practice and uses event-level GLMs. Each event regressor encodes the event's duration or fractional occupancy within each volume and is convolved with the canonical HRF. Thus, a brief stimulus occupying half of a 1-second volume contributes weight 0.5, whereas sustained events contribute positive weights, typically 1, across multiple successive volumes.

For each held-out condition or contrast, we fit identical design matrices to matched real and synthetic samples. This yields a real contrast map $\boldsymbol{\beta}_k$ and a synthetic contrast map $\hat{\boldsymbol{\beta}}_k$, which are compared across parcels. Real and synthetic analyses are always matched in sample size and event coverage, even when the dataset-specific windowing strategy does not cover every volume of the original run. For brevity, we use "contrast" to refer both to single event maps and to differences between event maps.

O.1 Contrast Inventory and Filtering

Filtering. The GLM inventory below includes only contrasts that passed a reliability check. Specifically, IBC contrasts with split-half reliability below $r = 0.25$ were excluded.

HCP contrasts. For HCP, contrasts are defined per task as follows:

EMOTION. face-shape perceptual blocks, plus shape against fixation.

GAMBLING. reward and loss feedback events in both directions.

LANGUAGE. story listening against math listening.

MOTOR. each movement block (left/right hand, left/right foot, tongue) against fixation.

RELATIONAL. relational reasoning blocks against shape-matching control blocks.

SOCIAL. mental-interaction movie blocks against random-motion movie blocks.

WM. 2-back working-memory blocks against 0-back blocks.

IBC contrast approach. IBC contrasts are task-specific linear combinations of event and, where needed, response regressors. Broad task-vs-baseline contrasts are used for tasks with a clear overall activation target; narrower task-vs-task contrasts are used when the experiment has a natural within-task comparison. Multi-condition contrasts average related regressors with equal or prespecified weights, while fixation/rest/silence terms are treated as baseline terms rather than standalone single conditions.

HCP contrast definitions.

EMOTION. Contrasts: `stim_face_gt_stim_shape = stim_face − stim_shape`; `stim_shape_gt_fixation = stim_shape − fixation`. Single conditions: `stim_face`, `stim_shape`.

GAMBLING. Contrasts: `reward_feedback_gt_loss_feedback = reward_feedback − loss_feedback`; `loss_feedback_gt_reward_feedback = loss_feedback − reward_feedback`. Single conditions: `reward_feedback`, `loss_feedback`.

LANGUAGE. Contrasts: `story_listen_gt_math_listen = story_listen − math_listen`. Single conditions: `story_listen`, `math_listen`.

MOTOR. Contrasts: `move_lh_gt_fixation = move_lh − fixation`; `move_rh_gt_fixation = move_rh − fixation`; `move_lf_gt_fixation = move_lf − fixation`; `move_rf_gt_fixation = move_rf − fixation`; `move_t_gt_fixation = move_t − fixation`. Single conditions: `move_lh`, `move_rh`, `move_lf`, `move_rf`, `move_t`.

RELATIONAL. Contrasts: `block_rel_gt_block_match = block_rel − block_match`. Single conditions: `block_rel`, `block_match`.

SOCIAL. Contrasts: `movie_mental_gt_movie_random = movie_mental − movie_random`. Single conditions: `movie_mental`, `movie_random`.

WM. Contrasts: `block_2bk_gt_block_0bk = block_2bk − block_0bk`. Single conditions: `block_2bk`, `block_0bk`.

HCP total: 13 contrasts and 17 single conditions.

IBC contrast definitions.

ArchiSocial. Contrasts: `social_gt_fixation = 0.25 events:false_belief_audio_story + 0.25 events:false_belief_video_story + 0.25 events:false_belief_why_prompt + 0.25 events:triangle_mental`. Single conditions: `events:false_belief_audio_story`, `events:false_belief_video_story`, `events:false_belief_why_prompt`, `events:triangle_mental`.

ArchiSpatial. Contrasts: `task_gt_fixation = 0.2 events:object_grasp + 0.2 events:object_orientation + 0.2 events:rotation_hand + 0.2 events:rotation_side + 0.2 events:saccade - events:fixation`. Single conditions: `events:object_grasp`, `events:object_orientation`, `events:rotation_hand`, `events:rotation_side`, `events:saccade`.

ArchiStandard. Contrasts: `sentences_gt_fixation = 0.5 events:audio_sentence + 0.5 events:video_sentence - events:fixation`. Single conditions: `events:audio_sentence`, `events:video_sentence`.

Audi. Contrasts: `all_audio_gt_silence = 0.08333 events:alphabet_audio + 0.08333 events:animal_sounds + 0.08333 events:coughing_sounds + 0.08333 events:crying_sounds + 0.08333 events:environmental_sounds + 0.08333 events:french_speech + 0.08333 events:human_non_speech_sounds + 0.08333 events:laughing_sounds + 0.08333 events:music_audio + 0.08333 events:reversed_speech + 0.08333 events:suomi_speech + 0.08333 events:yawning_sounds - events:silence_baseline`. Single conditions: `events:alphabet_audio`, `events:animal_sounds`, `events:coughing_sounds`, `events:crying_sounds`, `events:environmental_sounds`, `events:french_speech`, `events:human_non_speech_sounds`, `events:laughing_sounds`, `events:music_audio`, `events:reversed_speech`, `events:suomi_speech`, `events:yawning_sounds`.

BiologicalMotion1. Contrasts: `biological_motion_gt_fixation = 0.25 events:global_inverted + 0.25 events:global_upright + 0.25 events:natural_inverted + 0.25 events:natural_upright - events:fixation`; `global_gt_natural = 0.5 events:global_inverted + 0.5 events:global_upright - 0.5 events:natural_inverted - 0.5 events:natural_upright`. Single conditions: `events:global_inverted`, `events:global_upright`, `events:natural_inverted`, `events:natural_upright`.

BiologicalMotion2. Contrasts: `biological_motion_gt_fixation = 0.25 events:modified_inverted + 0.25 events:modified_upright + 0.25 events:natural_inverted + 0.25 events:natural_upright - events:fixation`. Single conditions: `events:modified_inverted`, `events:modified_upright`, `events:natural_inverted`, `events:natural_upright`.

BreathHolding. Contrasts: `hold_breath = events:hold_breath`; `hold_breath_gt_breathe = events:hold_breath - events:breathe`. Single conditions: `events:hold_breath`, `events:breathe`.

Catell. Contrasts: `oddball_gt_fixation = 0.5 events:easy_oddball + 0.5 events:hard_oddball - events:fixation`; `hard_oddball_gt_easy_oddball = events:hard_oddball - events:easy_oddball`. Single conditions: `events:easy_oddball`, `events:hard_oddball`.

Checkerboard. Contrasts: `checkerboard_stimulation_gt_fixation = events:checkerboard_stimulation - events:fixation`. Single conditions: `events:checkerboard_stimulation`.

Color. Contrasts: `stimulus_blocks_gt_fixation = 0.5 events:achromatic_block + 0.5 events:chromatic_block - events:fixation`. Single conditions: `events:achromatic_block`, `events:chromatic_block`.

ColumbiaCards. Contrasts: `task_gt_fixation = 0.3333 events:cards_risk_hi + 0.3333 events:cards_risk_lo + 0.3333 events:reward_feedback - events:fixation`. Single conditions: `events:cards_risk_hi`, `events:cards_risk_lo`, `events:reward_feedback`.

Discount. Contrasts: discount_trials_gt_fixation = 0.25 events:discount_amt_hi_del_late + 0.25 events:discount_amt_hi_del_soon + 0.25 events:discount_amt_lo_del_late + 0.25 events:discount_amt_lo_del_soon - events:fixation. Single conditions: events:discount_amt_hi_del_late, events:discount_amt_hi_del_soon, events:discount_amt_lo_del_late, events:discount_amt_lo_del_soon.

DotPatterns. Contrasts: all_stimuli = 0.25 events:cue_a + 0.25 events:cue_b + 0.25 events:probe_x + 0.25 events:probe_y. Single conditions: events:cue_a, events:cue_b, events:probe_x, events:probe_y.

EmoMem. Contrasts: object_overlay = events:object_overlay. Single conditions: events:object_overlay.

EmoReco. Contrasts: face_perception = 0.25 events:angry_female_face + 0.25 events:angry_male_face + 0.25 events:neutral_female_face + 0.25 events:neutral_male_face. Single conditions: events:angry_female_face, events:angry_male_face, events:neutral_female_face, events:neutral_male_face.

EmotionalPain. Contrasts: pain_story_gt_fixation = 0.5 events:emotional_pain_story + 0.5 events:physical_pain_story - events:fixation; emotional_pain_story_gt_physical_pain_story = events:emotional_pain_story - events:physical_pain_story. Single conditions: events:emotional_pain_story, events:physical_pain_story.

Enumeration. Contrasts: enumeration_gt_fixation = 0.125 events:enumeration_response_1 + 0.125 events:enumeration_response_2 + 0.125 events:enumeration_response_3 + 0.125 events:enumeration_response_4 + 0.125 events:enumeration_response_5 + 0.125 events:enumeration_response_6 + 0.125 events:enumeration_response_7 + 0.125 events:enumeration_response_8 - events:fixation; high_gt_low_numerosity_linear = -0.4375 events:enumeration_response_1 - 0.3125 events:enumeration_response_2 - 0.1875 events:enumeration_response_3 - 0.0625 events:enumeration_response_4 + 0.0625 events:enumeration_response_5 + 0.1875 events:enumeration_response_6 + 0.3125 events:enumeration_response_7 + 0.4375 events:enumeration_response_8. Single conditions: events:enumeration_response_1, events:enumeration_response_2, events:enumeration_response_3, events:enumeration_response_4, events:enumeration_response_5, events:enumeration_response_6, events:enumeration_response_7, events:enumeration_response_8.

FingerTap. Contrasts: fingertap_gt_fixation = events:fingertap - events:fixation. Single conditions: events:fingertap.

ItemRecognition. Contrasts: probe_digit_decision = events:probe_digit_decision; probe_digit_decision_gt_arrow_direction_decision = events:probe_digit_decision - events:arrow_direction_decision. Single conditions: events:probe_digit_decision, events:arrow_direction_decision.

Lec1. Contrasts: all_trials_gt_fixation = 0.3333 events:word_trial + 0.3333 events:pseudoword_trial + 0.3333 events:random_string_trial - events:fixation; word_gt_pseudoword_and_random_string = events:word_trial - 0.5 events:pseudoword_trial - 0.5 events:random_string_trial. Single conditions: events:word_trial, events:pseudoword_trial, events:random_string_trial.

MCSE. Contrasts: search_gt_fixation = 0.25 events:high_salience_left + 0.25 events:high_salience_right + 0.25 events:low_salience_left + 0.25 events:low_salience_right - events:fixation; low_salience_gt_high_salience = 0.5 events:low_salience_left + 0.5 events:low_salience_right - 0.5 events:high_salience_left - 0.5 events:high_salience_right. Single conditions: events:high_salience_left, events:high_salience_right, events:low_salience_left, events:low_salience_right.

MTTNS. Contrasts: all_events_gt_fixation = 0.125 events:after_close_event + 0.125 events:after_far_event + 0.125 events:before_close_event + 0.125 events:before_far_event + 0.125 events:northside_close_event + 0.125 events:northside_far_event + 0.125 events:southside_close_event + 0.125 events:southside_far_event - events:fixation; space_event_gt_time_event = 0.25 events:northside_close_event + 0.25 events:northside_far_event + 0.25 events:southside_close_event + 0.25 events:southside_far_event - 0.25 events:after_close_event - 0.25 events:after_far_event - 0.25 events:before_close_event - 0.25 events:before_far_event. Single conditions: events:after_close_event, events:after_far_event, events:before_close_event, events:before_far_event, events:northside_close_event, events:northside_far_event, events:southside_close_event, events:southside_far_event.

MTTWE. Contrasts: all_event_audio_gt_fixation = 0.25 events:after_event_audio + 0.25 events:before_event_audio + 0.25 events:eastside_event_audio + 0.25 events:westside_event_audio - events:fixation; time_event_audio_gt_space_event_audio = 0.5 events:after_event_audio + 0.5 events:before_event_audio - 0.5 events:eastside_event_audio - 0.5 events:westside_event_audio. Single conditions: events:after_event_audio, events:before_event_audio, events:eastside_event_audio, events:westside_event_audio.

MVEB. Contrasts: string_encoding_gt_fixation = 0.1667 events:two_letters_different_string + 0.1667 events:two_letters_same_string + 0.1667 events:four_letters_different_string + 0.1667 events:four_letters_same_string + 0.1667 events:six_letters_different_string + 0.1667 events:six_letters_same_string - events:fixation. Single conditions: events:two_letters_different_string, events:two_letters_same_string, events:four_letters_different_string, events:four_letters_same_string, events:six_letters_different_string, events:six_letters_same_string.

MVIS. Contrasts: task_gt_fixation = 0.125 events:empty_grid + 0.125 events:probe_dot_decision + 0.125 events:two_dots_array + 0.125 events:two_dots_control_array + 0.125 events:four_dots_array + 0.125 events:four_dots_control_array + 0.125 events:six_dots_array + 0.125 events:six_dots_control_array - events:fixation. Single conditions: events:empty_grid, events:probe_dot_decision, events:two_dots_array, events:two_dots_control_array, events:four_dots_array, events:four_dots_control_array, events:six_dots_array, events:six_dots_control_array.

MathLanguage. Contrasts: task_stimuli = 0.08333 events:arithmetic_fact_auditory + 0.08333 events:arithmetic_fact_visual + 0.08333 events:arithmetic_principle_auditory + 0.08333 events:arithmetic_principle_visual + 0.08333 events:colorlessg_auditory + 0.08333 events:colorlessg_visual + 0.08333 events:general_auditory + 0.08333 events:general_visual + 0.08333 events:geometry_fact_auditory + 0.08333 events:geometry_fact_visual + 0.08333 events:theory_of_mind_auditory + 0.08333 events:theory_of_mind_visual. Single conditions: events:arithmetic_fact_auditory, events:arithmetic_fact_visual, events:arithmetic_principle_auditory, events:arithmetic_principle_visual, events:colorlessg_auditory, events:colorlessg_visual, events:general_auditory, events:general_visual, events:geometry_fact_auditory, events:geometry_fact_visual, events:theory_of_mind_auditory, events:theory_of_mind_visual.

Moto. Contrasts: saccade_gt_other_movements = 0.5 events:saccade_left + 0.5 events:saccade_right - 0.125 events:finger_left - 0.125 events:finger_right - 0.125 events:foot_left - 0.125 events:foot_right - 0.125 events:hand_left - 0.125 events:hand_right - 0.125 events:tongue_left - 0.125 events:tongue_right. Single conditions: events:saccade_left, events:saccade_right, events:finger_left, events:finger_right, events:foot_left, events:foot_right, events:hand_left, events:hand_right, events:tongue_left, events:tongue_right.

NARPS. Contrasts: stim_gt_fixation = 0.25 events:narps_gamble_gain_hi_loss_hi + 0.25 events:narps_gamble_gain_hi_loss_lo + 0.25 events:narps_gamble_gain_lo_loss_hi + 0.25 events:narps_gamble_gain_lo_loss_lo - events:fixation. Single conditions: events:narps_gamble_gain_hi_loss_hi, events:narps_gamble_gain_hi_loss_lo, events:narps_gamble_gain_lo_loss_hi, events:narps_gamble_gain_lo_loss_lo.

PainMovie. Contrasts: movie_scenes_gt_fixation = 0.5 events:mental_movie_scene + 0.5 events:pain_movie_scene - events:fixation. Single conditions: events:mental_movie_scene, events:pain_movie_scene.

PreferenceFaces. Contrasts: face_evaluation_gt_fixation = 0.5 events:face_score_hi + 0.5 events:face_score_lo - events:fixation. Single conditions: events:face_score_hi, events:face_score_lo.

PreferenceFood. Contrasts: food_gt_fixation = 0.5 events:food_score_hi + 0.5 events:food_score_lo - events:fixation. Single conditions: events:food_score_hi, events:food_score_lo.

PreferenceHouses. Contrasts: houses_gt_fixation = 0.5 events:house_score_hi + 0.5 events:house_score_lo - events:fixation. Single conditions: events:house_score_hi, events:house_score_lo.

PreferencePaintings. Contrasts: paintings_gt_fixation = 0.5 events:painting_score_hi + 0.5 events:painting_score_lo - events:fixation. Single conditions: events:painting_score_hi, events:painting_score_lo.

RSVPLanguage. Contrasts: reading_blocks_gt_fixation = 0.1667 events:complex_sentence + 0.1667 events:simple_sentence + 0.1667 events:word_list + 0.1667 events:jabberwocky + 0.1667 events:pseudoword_list + 0.1667 events:consonant_strings - events:fixation. Single conditions: events:complex_sentence, events:simple_sentence, events:word_list, events:jabberwocky, events:pseudoword_list, events:consonant_strings.

SelectiveStopSignal. Contrasts: task_gt_fixation = 0.25 events:critical_go + 0.25 events:critical_stop + 0.25 events:noncritical_go + 0.25 events:noncritical_ignore - events:fixation; critical_stop_gt_noncritical_ignore = events:critical_stop - events:noncritical_ignore. Single conditions: events:critical_go, events:critical_stop, events:noncritical_go, events:noncritical_ignore.

Self. Contrasts: task_gt_fixation = 0.25 events:other_adjective_judgment + 0.25 events:self_adjective_judgment + 0.25 events:recognition_yes + 0.25 events:recognition_no - events:fixation. Single conditions: events:other_adjective_judgment, events:self_adjective_judgment, events:recognition_yes, events:recognition_no.

SpatialNavigation. Contrasts: encoding_phase = 0.5 events:encoding_intersection + 0.5 events:encoding_navigation; experimental_gt_control = events:experimental_approach - events:control_approach. Single conditions: events:encoding_intersection, events:encoding_navigation, events:experimental_approach, events:control_approach.

StopSignal. Contrasts: task_trials = 0.5 events:go_trial + 0.5 events:stop_trial; stop_trial_gt_go_trial = events:stop_trial - events:go_trial. Single conditions: events:go_trial, events:stop_trial.

Stroop. Contrasts: task_all = 0.5 events:congruent + 0.5 events:incongruent. Single conditions: events:congruent, events:incongruent.

TheoryOfMind. Contrasts: task_gt_fixation = 0.25 events:belief_story + 0.25 events:belief_question + 0.25 events:photo_story + 0.25 events:photo_question - events:fixation; belief_gt_photo = 0.5 events:belief_story + 0.5 events:belief_question - 0.5 events:photo_story - 0.5 events:photo_question. Single conditions: events:belief_story, events:belief_question, events:photo_story, events:photo_question.

TwoByTwo. Contrasts: task_stimulus_gt_fixation = events:task_stimulus - events:fixation. Single conditions: events:task_stimulus.

VSTM. Contrasts: all_stimuli_gt_fixation = 0.08333 events:vstm_encode_load_1 + 0.08333 events:vstm_encode_load_2 + 0.08333 events:vstm_encode_load_3 + 0.08333 events:vstm_encode_load_4 + 0.08333 events:vstm_encode_load_5 + 0.08333 events:vstm_encode_load_6 + 0.08333 events:vstm_probe_load_1 + 0.08333 events:vstm_probe_load_2 + 0.08333 events:vstm_probe_load_3 + 0.08333 events:vstm_probe_load_4 + 0.08333 events:vstm_probe_load_5 + 0.08333 events:vstm_probe_load_6 - events:fixation. Single conditions: events:vstm_encode_load_1, events:vstm_encode_load_2, events:vstm_encode_load_3, events:vstm_encode_load_4, events:vstm_encode_load_5, events:vstm_encode_load_6, events:vstm_probe_load_1, events:vstm_probe_load_2, events:vstm_probe_load_3, events:vstm_probe_load_4, events:vstm_probe_load_5, events:vstm_probe_load_6.

VSTMC. Contrasts: stim = 0.3333 events:stimulus_load1 + 0.3333 events:stimulus_load2 + 0.3333 events:stimulus_load3. Single conditions: events:stimulus_load1, events:stimulus_load2, events:stimulus_load3.

VisualSearch. Contrasts: task_stimuli = 0.09091 events:sample_item + 0.09091 events:memory_array_two + 0.09091 events:memory_array_four + 0.09091 events:probe_item_two_absent + 0.09091 events:probe_item_two_present + 0.09091 events:probe_item_four_absent + 0.09091 events:probe_item_four_present + 0.09091 events:search_array_two_absent + 0.09091 events:search_array_two_present + 0.09091 events:search_array_four_absent + 0.09091 events:search_array_four_present. Single conditions: events:sample_item, events:memory_array_two, events:memory_array_four, events:probe_item_two_absent, events:probe_item_two_present, events:probe_item_four_absent, events:probe_item_four_present, events:search_array_two_absent, events:search_array_two_present, events:search_array_four_absent, events:search_array_four_present.

WardAndAllport. Contrasts: planning_problem_gt_fixation = events:planning_problem - events:fixation; planning_problem_gt_move_execution = events:planning_problem - events:move_execution. Single conditions: events:planning_problem, events:move_execution.

IBC total: 60 contrasts and 191 single conditions.

P LLM Use

We use LLMs to generate and rephrase captions of task instructions, sensory presentations, and responses. A mixture of GPT-5.4-codex by OpenAI and Claude Opus 4.6 by Anthropic were used for this purpose.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the model, conditioning regimes, zero-shot scope, and intended use cases. The Methods and Results sections evaluate these claims on held-out tasks across contrast recovery, distance-to-support analyses, time-series statistics, and counterfactual timing experiments.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper includes a Limitations section discussing the scope of extrapolation, assumptions behind compositional generalization, oracle-like spatial priors, and limits of language-only conditioning.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper uses mathematical notation to define the model and flow-matching objective, but it does not introduce new theoretical results, theorems, or proofs

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The Methods and Experimental Setup sections describe the architecture, conditioning tokens, flow-matching objective, datasets, preprocessing, held-out folds, GLM evaluation, baselines, hyperparameters, and compute budget; additional optimization, dropout, sampling, and evaluation details are referenced in the appendix. There we also describe how to access the data used.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The paper uses controlled-access neuroimaging datasets whose raw data must be requested from the original providers under dataset-specific data-use agreements. We provide data-access information and implementation details in the supplement, but do not include a fully reproducible code and data release at submission time. We will release the training/evaluation code, preprocessing scripts, and model weights upon acceptance, subject to the redistribution terms of the original datasets.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The main paper specifies the datasets, held-out-task and held-out-subject splits, model architecture, conditioning pathways, evaluation protocol, main hyperparameters, and baselines needed to interpret the results. Additional optimization details, sampling weights, dropout probabilities, contrast definitions, and dataset-specific evaluation details are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: For all instances where errors are reported, we clarify that they are standard deviations. We use statistical testing to support two of our claims and indicate the test and over which data they were computed.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We include the compute use in its own Appendix section and explicitly refer to this in the main text.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We confirm to have read and conducted our research fully in line with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses positive impacts for in-silico task design and counterfactual neuroscience, as well as limitations and risks related to over-interpreting generated brain responses, subject-identifiable neuroimaging structure, and the need for controlled use of individual-level data.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release a high-risk model or dataset at submission time. Raw neuroimaging data remain governed by the original providers' controlled-access agreements; any future release of code, weights, or generated data will follow these terms.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites all existing datasets and model components used. Dataset access is governed by the respective providers' data-use agreements; raw data are not redistributed. Code and model dependencies will be documented in the release package.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: No new public dataset, code package, or model weights are released with the initial submission. Upon acceptance, we plan to release code, configuration files, preprocessing scripts, and trained model weights, subject to the redistribution terms of the source datasets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not conduct new crowdsourcing experiments or collect new human-subject data. It analyzes previously collected neuroimaging datasets

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not collect new human-subject data. All analyses use previously collected datasets accessed under their respective data-use agreements and original ethical approvals.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The method uses an LLM to generate phrase variants of task captions for conditioning robustness. The paper describes this use in the appendix. LLMs were not used to generate experimental results beyond these caption variants.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.