

Supplementary Information for: Decoding Future Semantics from Language Model Hidden States: Rollout Trade-offs and Causal Steering

Neural Processing Letters

Jaekeol Choi^{1*}

^{1*}Division of AI Data Convergence, Hankuk University of Foreign
Studies, Yongin-si, Republic of Korea.

Corresponding author(s). E-mail(s): jaekeol.choi@hufs.ac.kr;

Contents of Online Resource 1

This document contains additional semantic-probe results, robustness analyses, causal-steering details, and reproducibility information supporting the main manuscript.

Appendix A Additional Semantic Probe Results

This appendix provides additional results supporting the passive semantic-recoverability analysis in the main manuscript. The main text reports the most informative comparisons, while the following subsections provide the underlying layer-wise probe results, complete representation-level baseline comparisons, and retrieval-based evaluation. Steering-specific layer sweeps are reported separately in Appendix 3.

A.1 Full Layer-wise Probe Results

Table A1 reports the passive semantic-probe performance across the predefined layers of the Pythia-1.4B anchor model using MiniLM continuation embeddings. The probe uses the hidden state at the final prompt token and predicts the semantic representation of the model’s own continuation through linear ridge regression.

Semantic recoverability increases gradually from layers 4 to 12 and then rises more substantially at layer 20. The best tested layer improves mean cosine by 0.0920 over

Table A1 Layer-wise passive semantic recoverability for Pythia-1.4B using MiniLM targets.

Layer	Probe cosine	Δ vs. layer 4
4	0.3363	—
8	0.3412	+0.0049
12	0.3670	+0.0307
20	0.4283	+0.0920

layer 4, corresponding to approximately a 27% relative increase. This pattern motivated the use of layer 20 as the primary passive readout location for the Pythia-1.4B anchor experiments.

The passive layer profile should not, however, be interpreted as identifying the optimal layer for causal intervention. The broader steering-specific sweep includes additional layers and evaluates causal effects independently; those results are reported in Appendix 3.

A.2 Representation-Level Baselines

Table A2 provides the representation-level controls used to evaluate whether the observed semantic recoverability can be explained by simpler properties of the continuation space or by token-level information available from the same hidden state. All results use the Pythia-1.4B anchor model and MiniLM continuation embeddings.

Table A2 Representation-level baselines for Pythia-1.4B. The margin is computed relative to the layer-20 semantic probe.

Method	Cosine	Probe margin
Semantic probe	0.4283	—
Mean embedding	0.2310	+0.1973
Random match	0.0616	+0.3667
Lexical TF-IDF	0.3612	+0.0671
Best logit-lens control	0.1812	+0.2471

The mean-embedding baseline measures the extent to which a single central point in the continuation embedding space can obtain non-trivial cosine similarity without using prompt-specific information. Its score of 0.2310 is substantially below the semantic probe. The random-match baseline is lower still at 0.0616, indicating that the probe’s performance cannot be explained by arbitrary similarity between unrelated prompt–continuation pairs.

The lexical TF-IDF baseline is substantially stronger, reaching 0.3612. This result shows that prompt surface information alone carries meaningful predictive signal about the continuation semantics. Nevertheless, the semantic probe remains higher by 0.0671

cosine, indicating that the hidden representation contains information beyond that obtainable from simple lexical features.

The logit-lens controls test whether the probe advantage can instead be explained by token identities already favored by the model at the current state. Six predefined variants were evaluated by combining

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

with concatenation- and probability-weighted aggregation schemes. The strongest of these variants achieves a cosine similarity of only 0.1812, leaving a margin of 0.2471 in favor of the semantic probe. Thus, directly converting the current token-level preferences into a semantic representation does not account for the observed continuation-level recoverability.

These controls support the interpretation that the semantic probe captures prompt-specific information present in the hidden representation that is not reducible to the global continuation distribution, prompt lexical content, or a simple readout of currently preferred token identities. As shown in the main manuscript, however, these representation-level controls are substantially weaker than explicit behavioral rollout, motivating the stronger behavioral-rollout comparison reported there.

A.3 Retrieval Evaluation

To complement cosine-based evaluation, the predicted continuation embeddings are also evaluated as retrieval queries. For each test prompt, the probe prediction is compared against all continuation embeddings in the held-out test set, and the continuation generated from the same prompt is treated as the relevant item. This evaluation tests whether the predicted representation is sufficiently specific to distinguish the correct future continuation from other semantically plausible outputs.

Table A3 reports the retrieval results for the Pythia-1.4B anchor model with MiniLM targets. The probe retrieves the correct continuation at rank 1 for 39.76% of test examples and within the top five for 73.49%. The mean rank of the correct continuation is 5.65.

Table A3 Retrieval performance of the Pythia-1.4B semantic probe using MiniLM continuation embeddings. Chance values assume uniform ranking among the 83 held-out candidates.

Metric	Probe	Chance
Recall@1	0.3976	0.0120
Recall@5	0.7349	0.0602
Mean rank	5.65	42.00

The large gap from chance indicates that the predicted embeddings are not merely located in a generally dense region of the semantic space. Instead, they frequently identify the specific continuation associated with the current prompt. In particular, Recall@5 is more than twelve times the uniform-chance level, while the mean rank improves from the chance expectation of 42.0 to 5.65.

These retrieval results provide a complementary perspective to cosine similarity. Whereas cosine measures the absolute proximity between a predicted and true semantic embedding, retrieval evaluates the prediction relative to competing continuations. The strong performance under both criteria supports the conclusion that the current hidden state contains prompt-specific information about the semantics of the model’s future output.

Appendix B Additional Robustness Analyses

This appendix provides supplementary results for the encoder-robustness and horizon-rollout analyses reported in the main manuscript. It includes detailed semantic-encoder results, repeated-split sensitivity of the prompt-semantic controls, and full probe-rollout comparisons across horizons and semantic target spaces.

B.1 Full Semantic-Encoder Robustness Results

Table ?? summarizes the Pythia-1.4B results for MiniLM, BGE-base, and E5-base. Layer 20 is the best passive probe layer for all three encoders.

Absolute cosine values vary substantially across encoders because their embedding geometries differ. The relevant comparison is therefore within each semantic space. The probe outperforms the corresponding weak and token-level controls for all three encoders, and shuffled-target training substantially reduces recoverability, confirming that the effect depends on prompt-specific hidden-state information.

The rollout comparison is more encoder-dependent. A three-token rollout already exceeds the MiniLM probe, whereas the probe remains stronger than rollout-3 for BGE-base and E5-base. At five tokens, rollout clearly exceeds the probe for MiniLM and BGE-base but is approximately at parity with E5-base. Thus, semantic recoverability is robust across encoders, while the rollout budget required to match the probe depends on the semantic target space.

2.2 Repeated-Split Robustness of Prompt-Semantic Controls

To assess sensitivity to the particular train-validation-test partition used in the primary analysis, the prompt-semantic control experiment is repeated over ten alternative 383/83/83 splits. For each split and semantic encoder, ridge regularization is reselected on the corresponding validation set. Table B5 reports the test-set advantage of the hidden-state probe over Prompt Ridge.

The MiniLM advantage remains positive across all ten alternative splits and is consistently larger than the corresponding BGE and E5 differences. For BGE and E5, the advantage is close to zero and becomes slightly negative in one split, indicating greater sensitivity to partition choice. These results complement the paired-bootstrap

Table B5 Repeated-split robustness of the hidden-state probe advantage over Prompt Ridge. Each entry is the test cosine difference $S_{\text{hidden}} - S_{\text{prompt}}$. The splits substantially overlap and are therefore treated as a descriptive sensitivity analysis rather than as independent replications.

Split seed	MiniLM	BGE-base	E5-base
1	0.0224	0.0049	0.0022
2	0.0264	0.0025	0.0012
3	0.0403	0.0074	0.0032
4	0.0283	0.0060	0.0021
5	0.0245	0.0052	0.0021
6	0.0336	0.0069	0.0021
7	0.0273	0.0036	0.0011
8	0.0299	0.0070	0.0022
9	0.0167	-0.0037	-0.0005
10	0.0304	0.0071	0.0027
Mean $\pm$ std	0.0280 ± 0.0064	0.0047 ± 0.0033	0.0018 ± 0.0010

Table B6 Probe-rollout margins across prediction horizons and semantic encoders. Each cell reports mean $\Delta(H, m, E)$ with its paired-bootstrap 95% confidence interval based on 10,000 resamples. All conditions use the same 87 held-out test examples.

Encoder	H	$m = 3$		$m = 5$		$m = 10$		$m = 20$	
MiniLM	16	0.0446	[-0.0028, 0.0913]	-0.1321	[-0.1806, -0.0843]	-0.3845	[-0.4203, -0.3484]	-0.4887	[-0.5197, -0.4577]
	48	0.1260	[0.0813, 0.1693]	-0.0231	[-0.0743, 0.0263]	-0.2444	[-0.2853, -0.2038]	-0.4296	[-0.4660, -0.3937]
	96	0.1433	[0.1003, 0.1854]	0.0120	[-0.0384, 0.0608]	-0.1820	[-0.2251, -0.1402]	-0.3654	[-0.4052, -0.3256]
	192	0.1844	[0.1434, 0.2235]	0.0756	[0.0278, 0.1210]	-0.0953	[-0.1374, -0.0541]	-0.2661	[-0.3060, -0.2265]
	256	0.2062	[0.1661, 0.2450]	0.1036	[0.0566, 0.1480]	-0.0576	[-0.1003, -0.0159]	-0.2228	[-0.2625, -0.1834]
BGE-base	16	0.0572	[0.0339, 0.0801]	-0.0383	[-0.0609, -0.0155]	-0.1903	[-0.2065, -0.1747]	-0.2490	[-0.2633, -0.2351]
	48	0.1114	[0.0927, 0.1298]	0.0359	[0.0158, 0.0554]	-0.0831	[-0.0990, -0.0671]	-0.1886	[-0.2025, -0.1747]
	96	0.1326	[0.1146, 0.1504]	0.0643	[0.0441, 0.0838]	-0.0396	[-0.0565, -0.0227]	-0.1381	[-0.1527, -0.1239]
	192	0.1549	[0.1363, 0.1726]	0.0889	[0.0683, 0.1090]	-0.0096	[-0.0280, 0.0086]	-0.1015	[-0.1171, -0.0857]
	256	0.1624	[0.1438, 0.1804]	0.0973	[0.0763, 0.1175]	0.0012	[-0.0175, 0.0197]	-0.0896	[-0.1059, -0.0730]
E5-base	16	0.0515	[0.0424, 0.0604]	0.0087	[-0.0008, 0.0181]	-0.0712	[-0.0788, -0.0638]	-0.1049	[-0.1109, -0.0988]
	48	0.0824	[0.0750, 0.0897]	0.0520	[0.0433, 0.0604]	-0.0030	[-0.0113, 0.0051]	-0.0701	[-0.0768, -0.0634]
	96	0.0888	[0.0811, 0.0962]	0.0612	[0.0520, 0.0700]	0.0119	[0.0031, 0.0207]	-0.0515	[-0.0585, -0.0443]
	192	0.0918	[0.0845, 0.0988]	0.0653	[0.0562, 0.0743]	0.0211	[0.0124, 0.0298]	-0.0348	[-0.0415, -0.0281]
	256	0.0941	[0.0870, 0.1009]	0.0692	[0.0605, 0.0779]	0.0279	[0.0193, 0.0365]	-0.0263	[-0.0328, -0.0196]

encoder contrasts in the main manuscript without treating the overlapping splits as independent experimental replications.

2.3 Full Horizon-Rollout Results

The fixed-horizon analysis evaluates $H \in \{16, 48, 96, 192, 256\}$ and rollout budgets $m \in \{3, 5, 10, 20\}$ under MiniLM, BGE-base, and E5-base. All conditions use the same 87 held-out test examples. Across all 12 encoder-budget series, the observed mean probe-rollout margin increases monotonically with horizon.

Table B6 reports the full margins and paired-bootstrap 95% confidence intervals. Negative values indicate an advantage for rollout.

The observed rollout parity budget is non-decreasing with horizon for all three semantic encoders. Table B7 summarizes the smallest tested rollout budget whose mean semantic similarity matches or exceeds the probe.

The $m = 20$ rollout remains stronger in every encoder-horizon condition, with all corresponding confidence intervals below zero. These results support the descriptive

Table B7 Smallest tested rollout budget whose mean semantic similarity matches or exceeds the semantic probe.

Horizon H	MiniLM	BGE-base	E5-base
16	5	5	10
48	5	10	10
96	10	10	20
192	10	10	20
256	10	20	20

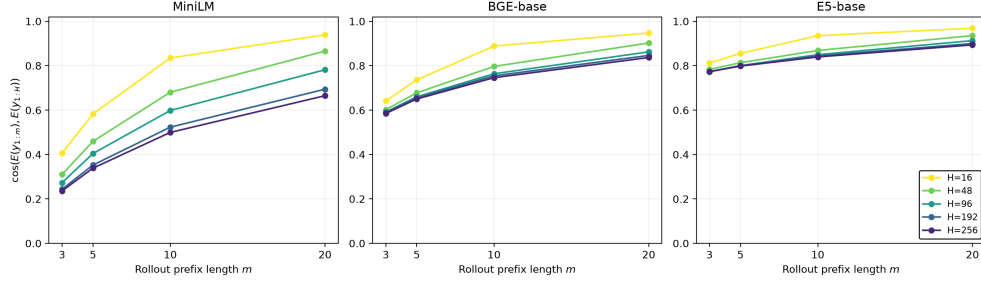


Fig. B1 Prefix-to-full semantic convergence across MiniLM, BGE-base, and E5-base. The x-axis denotes rollout prefix length m , and each curve corresponds to a prediction horizon H . Values are cosine similarities between the embedding of the generated prefix $y_{1:m}$ and that of the full continuation $y_{1:H}$.

horizon trend while also showing that near-parity conditions at smaller rollout budgets can have confidence intervals that include zero.

2.4 Prefix-to-Full Semantic Convergence

Figure B1 shows how the semantic representation of a short generated prefix approaches that of the full continuation as the rollout length increases. The convergence pattern differs across semantic encoders, consistent with the encoder-dependent rollout parity budgets reported in the main manuscript.

Longer prefixes consistently become more similar to the full continuation, but the rate of convergence varies across embedding spaces and horizons. This variation explains why the rollout budget required to match the hidden-state probe is not fixed across semantic encoders.

2.5 Marginal Return of Rollout Computation

Figure B2 shows the marginal semantic gain per additional rollout token. For each encoder and prediction horizon, the gain decreases as the rollout budget increases.

The consistently diminishing returns indicate that early rollout tokens contribute more semantic information than later ones. This trend complements the horizon analysis by showing that increasing rollout computation remains beneficial, but its semantic benefit per additional token progressively decreases.

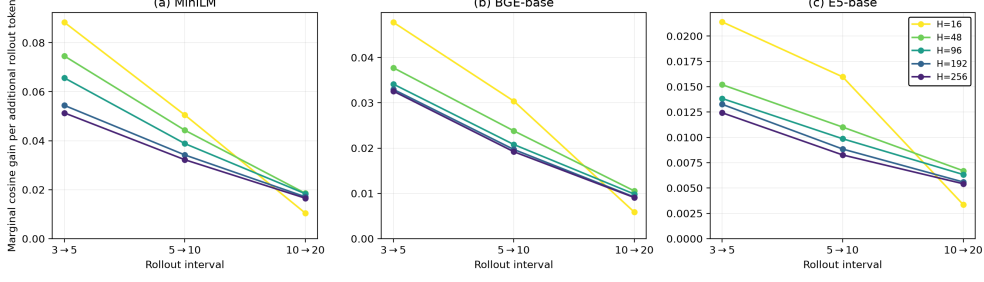


Fig. B2 Marginal semantic gain per additional rollout token across MiniLM, BGE-base, and E5-base. Each curve corresponds to a prediction horizon H , and the x-axis denotes consecutive rollout-budget intervals.

3 Causal Steering Details

This appendix provides supplementary details and full layer-wise results for the causal steering experiments. The emphasis is on the construction and scaling of probe-derived interventions and on the variation of steering effectiveness across layers and model families.

3.1 Steering Direction and Intervention Scale

The finalized steering experiments use a single sentiment contrast defined in the MiniLM semantic space. The five positive anchors are “This is wonderful.”, “The outcome was excellent.”, “It made everyone happy.”, “Everything turned out great.”, and “What a fantastic result.” The five negative anchors are “This is terrible.”, “The outcome was awful.”, “It made everyone upset.”, “Everything turned out badly.”, and “What a disastrous result.”

Each anchor is encoded with L2 normalization. Let

$$\bar{\mathbf{e}}_+ = \frac{1}{5} \sum_{i=1}^5 E(a_i^+), \quad \bar{\mathbf{e}}_- = \frac{1}{5} \sum_{i=1}^5 E(a_i^-).$$

The unit sentiment axis is

$$\mathbf{u} = \frac{\bar{\mathbf{e}}_+ - \bar{\mathbf{e}}_-}{\|\bar{\mathbf{e}}_+ - \bar{\mathbf{e}}_-\|_2}.$$

For layer ℓ , the corresponding hidden-state direction is obtained from the ridge probe refit on the combined training and validation splits:

$$\mathbf{d}_\ell = \frac{W_\ell^\top \mathbf{u}}{\|W_\ell^\top \mathbf{u}\|_2}.$$

The intervention magnitude is normalized by the validation-set hidden-state norm,

$$\alpha_\ell = k_\ell \cdot \mathbb{E}_{x \in \mathcal{D}_{\text{val}}} [\|h_\ell(x)\|_2],$$

where k_ℓ is selected on validation-pool pilot prompts and fixed before final test evaluation. The candidate grids and layer-specific scale-selection procedure are detailed in Appendix 4.1.

The vector $\alpha_\ell \mathbf{d}_\ell$ is added once during prompt prefill at the final prompt-token position. Under the `hidden_states` indexing convention used throughout the paper, hidden-state index ℓ corresponds to the output residual stream of transformer block $\ell - 1$. No intervention is applied during subsequent autoregressive decoding steps.

Generated continuations are scored using normalized positive and negative centroids,

$$s(y) = \cos(E(y), \hat{\mathbf{e}}_+) - \cos(E(y), \hat{\mathbf{e}}_-),$$

where

$$\hat{\mathbf{e}}_+ = \frac{\bar{\mathbf{e}}_+}{\|\bar{\mathbf{e}}_+\|_2}, \quad \hat{\mathbf{e}}_- = \frac{\bar{\mathbf{e}}_-}{\|\bar{\mathbf{e}}_-\|_2}.$$

For each evaluated prompt, the sentiment and matched-random intervention effects are measured relative to the unsteered generation:

$$S_{\text{sem},\ell} = s(y_\ell^{\text{sem}}) - s(y^{\text{base}}), \quad S_{\text{rand},\ell} = s(y_\ell^{\text{rand}}) - s(y^{\text{base}}),$$

and the steering margin is

$$M_\ell = S_{\text{sem},\ell} - S_{\text{rand},\ell}.$$

For the matched control, a Gaussian random direction is independently sampled for each prompt and evaluation seed, L2-normalized, and scaled by the same α_ℓ as the sentiment direction. Final evaluation uses five evaluation seeds. For each seed, 60 eligible prompts are sampled from the held-out steering test pool, and matched random directions are independently resampled; the probe-derived sentiment direction, intervention magnitude, and greedy generation procedure remain fixed. The steering margin is averaged within each evaluation seed, and the reported 95% intervals are Student- t intervals over the five seed-level means, following the evaluation protocol described in the main manuscript. Specifically, the interval is

$$\bar{M}_\ell \pm t_{0.975,4} \frac{s_\ell}{\sqrt{5}},$$

where s_ℓ is the standard deviation of the five seed-level mean margins. These intervals summarize replicate-level variation from prompt subsampling and matched random-control sampling rather than uncertainty from probe retraining or training-split resampling.

3.2 Cross-Model Steering Summary

Table ?? reports the final sentiment-steering results across all evaluated models. For Pythia-410M and Pythia-2.8B, only the passive-best tested layer was evaluated for steering; the remaining models received steering-specific layer sweeps.

3.3 Layer-wise Steering in Pythia Models

Table ?? reports the full steering-specific layer sweeps for Pythia-1.0B and Pythia-1.4B. The steering margin subtracts the matched random-direction control, and the reported 95% intervals are Student- t intervals over five evaluation-seed means.

The two models exhibit different relationships between passive and causal layer profiles. For Pythia-1.0B, the best passive layer is 13, whereas the largest observed steering margin occurs at layer 6. In contrast, Pythia-1.4B reaches both its best passive recoverability and largest observed steering margin at layer 20. The negative margins at Pythia-1.0B layer 4 and Pythia-1.4B layer 16 have replicate-level intervals entirely below zero, further showing that strong semantic decodability does not guarantee that intervention along the corresponding direction will move generation in the intended direction.

3.4 Layer-wise Steering in Qwen Models

Table ?? summarizes the layer-wise sentiment-steering sweeps for the Qwen models, together with the passive-best layer and the best-performing tested steering layer under the pilot-selected intervention scales.

The Qwen models show different relationships between passive semantic decodability and sentiment-steering effectiveness. Qwen2.5-1.5B and Qwen3-1.7B exhibit clear separation between the passive-best and best-performing tested steering layers, whereas Qwen3-4B places them in adjacent layers. Together with the Pythia results, these observations show that the layer providing the strongest passive semantic read-out is not necessarily the layer yielding the largest observed sentiment-steering margin under the evaluated pilot-selected intervention scales.

4 Steering Validation and Reproducibility

This appendix provides additional validation of the causal steering protocol, focusing on intervention-scale normalization, multi-seed robustness, and computational reproducibility.

4.1 Intervention-Scale Selection and Normalization

Using the same absolute intervention scale across models and layers can produce misleading comparisons because hidden-state norms differ substantially. Steering strength is therefore defined relative to the typical activation magnitude at each layer:

$$\alpha_\ell = k_\ell \cdot \mathbb{E}_{x \in \mathcal{D}_{\text{val}}} [\|\mathbf{h}_\ell(x)\|_2]. \quad (\text{D1})$$

For each evaluated layer, k_ℓ is selected using 40 eligible pilot prompts drawn from the validation split and disjoint from the final test split. The selected coefficient maximizes the mean pilot steering margin between the probe-derived semantic intervention and its matched random-direction control, and is fixed before final test evaluation.

Table D11 Pilot-selected intervention-scale coefficients. Layers follow the Hugging Face `hidden_states` indexing convention. Coefficients within each row correspond to the listed layers in order.

Model	Grid	Layers	Selected k_ℓ
Pythia-410M	Sparse	20	6.4
Pythia-1.0B	Dense	1, 4, 6, 9, 11, 13, 16	1.0, 2.4, 8.0, 8.0, 1.6, 0.2, 3.2
Pythia-1.4B	Dense	1, 4, 8, 12, 16, 20, 24	1.6, 0.1, 1.0, 0.2, 1.0, 8.0, 8.0
Pythia-2.8B	Sparse	27	3.2
Qwen2.5-1.5B	Dense	4, 10, 14, 24, 28	4.8, 0.1, 1.6, 8.0, 4.8
Qwen3-1.7B	Dense	4, 10, 14, 20, 24, 28	0.4, 6.4, 6.4, 1.6, 1.0, 6.4
Qwen3-4B	Dense	5, 13, 18, 26, 31, 32, 36	8.0, 3.2, 1.6, 4.8, 8.0, 3.2, 6.4

Two predefined scale grids are used. The initial passive-layer evaluations use the sparse grid

$$\{0.005, 0.01, 0.02, 0.04, 0.08, 0.16, 0.4, 0.8, 1.6, 3.2, 6.4\},$$

whereas the subsequent layer-wise sweeps use the denser grid

$$\{0.05, 0.1, 0.2, 0.3, 0.4, 0.6, 0.8, 1.0, 1.6, 2.4, 3.2, 4.8, 6.4, 8.0\}.$$

Table D11 summarizes the coefficients used in the finalized evaluations.

Initial experiments using an unnormalized fixed α produced unstable cross-model steering effects, motivating the norm-relative formulation used in the finalized analyses. Because k_ℓ is selected separately for each tested layer, layer-wise steering comparisons should be interpreted as performance under independently pilot-selected intervention scales rather than under a single globally fixed perturbation magnitude.

4.2 Multi-Seed Validation of Steering Effects

Several preliminary steering results changed materially after norm-relative scale normalization, replicated evaluation, and steering-specific layer sweeps. Table ?? summarizes two representative cases.

For Pythia-1.0B, evaluation only at the passive-best layer suggested essentially no sentiment-steering advantage over the matched random control, whereas the steering-specific sweep yielded a substantially larger observed margin at layer 6. For Pythia-2.8B, the near-zero preliminary result under a fixed raw intervention scale was replaced by a small positive mean margin under the finalized norm-relative protocol, with its replicate-level interval remaining above zero. The finalized evaluation uses five evaluation seeds, across which both the sampled held-out prompts and matched random directions vary. These cases illustrate why steering conclusions should not rely on a single evaluation realization or assume that the best passive readout layer is also the best intervention layer.

4.3 Computational Cost and Artifact Reuse

All experiments were implemented in Python using PyTorch and the Hugging Face `transformers` ecosystem, with `sentence-transformers` for semantic target encoders and `scikit-learn` for linear probing. Language models were loaded in `float16`

and evaluated on a single NVIDIA RTX PRO 6000 Blackwell GPU. The complete experimental suite required approximately three GPU-hours.

Unless otherwise noted, continuation generation is greedy (`do_sample=False`, `num_beams=1`). The primary variable-length protocol generates at most 24 new tokens, applies EOS and first-sentence truncation, and excludes continuations shorter than five tokens. The fixed-horizon protocol generates up to 256 new tokens without sentence-boundary truncation. Continuations reaching the full 256-token horizon form the common 577-example subset used for all fixed-horizon comparisons. The sampling-robustness analysis uses temperature 0.8 and $\text{top-}p = 0.95$, followed by the same post-processing as the primary protocol.

To avoid redundant computation, analyses reused cached continuations, hidden states, and semantic embeddings whenever the corresponding quantities were unchanged by the experimental condition. Semantic-encoder robustness reused the same hidden states and continuations with alternative sentence encoders, while the fixed-horizon and latent-decoding analyses reused previously generated continuation trajectories. The Pythia-1.4B steering-layer sweep likewise reused compatible base-model artifacts while evaluating the required layer-specific interventions separately.

Artifact reuse preserves the original underlying train/validation/test partitions and continuation targets; only the analysis or intervention specific to each experiment is recomputed. This substantially reduces supplementary computational cost without changing the evaluation data.

All hyperparameter and intervention-scale selection used training, validation, or held-out pilot data only. Test examples were reserved for final evaluation, and steering-specific intervention magnitudes were fixed before final multi-seed testing. Complete source code, exact checkpoint identifiers, and experiment configurations are available at https://github.com/maygodwithu/decoding_future_semantic.git.

4.4 Probe and Latent-Decoder Configuration

Table D13 summarizes the implementation details of the semantic probes and latent-decoding baselines. Unless otherwise noted, hyperparameters are selected using the validation split and are fixed before test evaluation.

The Tuned-Lens-style baseline should be interpreted as a controlled affine next-token decoder trained within the available experimental split, rather than as a reproduction of a publicly pretrained Tuned Lens checkpoint. Its translator is optimized and selected using next-token cross-entropy, whereas the reported downstream comparison uses continuation-level semantic cosine similarity. The result therefore characterizes this tested decoder configuration rather than establishing a general performance ordering between Tuned Lens and direct logit-lens methods.

Table D13 Configuration of the semantic probes and latent-decoding baselines. “Default” denotes a library default rather than an explicitly specified hyperparameter.

Component	Configuration
Ridge semantic probe	<code>sklearn.linear_model.Ridge</code> with $\alpha \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3, 10^4, 10^5, 10^6\}$. The solver is <code>auto</code> and <code>fit_intercept=True</code> (both library defaults). Hidden states are used without explicit centering, standardization, or L2 normalization. Semantic targets are L2-normalized before fitting. For each layer, α is selected by validation mean cosine similarity, after which the layer is selected using the same validation criterion.
MLP semantic probe	<code>sklearn.neural_network.MLPRegressor</code> with one 512-unit ReLU hidden layer and no dropout. The input dimension is 2048 and the output dimension is 384 for the Pythia-1.4B/MiniLM comparison. The optimizer is Adam with default learning rate 10^{-3} and default automatic batch size (200 for 403 training examples). The regularization parameter is selected from $\{0.01, 0.1, 1.0\}$ by validation cosine similarity. Training uses at most 800 iterations, with the default <code>tol</code> = 10^{-4} and <code>n_iter_no_change</code> =10. Inputs are not normalized; targets are L2-normalized.
Direct logit lens	Top- k variants with $k \in \{1, 5, 10\}$ are obtained from softmax probabilities. Candidate token IDs are individually decoded and combined at the text level using either probability-ranked concatenation or probability-proportional repetition. The resulting text is embedded once with MiniLM. The best of the six $k \times$ aggregation variants is selected by validation cosine similarity.
Tuned-Lens-style affine translator	A custom affine translator is trained separately at candidate layers 4, 8, 12, and 20; no pretrained Tuned Lens checkpoint or external training corpus is used. The language model is frozen. Each translator is trained on the 403 fixed-horizon training examples to predict the first continuation token using cross-entropy, Adam ($\text{lr} = 10^{-3}$, weight decay = 10^{-4}), full-batch optimization, at most 300 epochs, and patience 15 based on validation cross-entropy. At evaluation, the same top- k and text-aggregation variants as the direct logit lens are used, with the final variant selected by validation semantic cosine similarity.
Future-Lens-style decoder	Ten independent affine translators predict offsets $1, \dots, 10$ from the same pre-generation layer-20 hidden state. Each is trained with cross-entropy using Adam ($\text{lr} = 10^{-3}$, weight decay = 10^{-4}), full-batch optimization, at most 300 epochs, and patience 15 based on validation cross-entropy. For rollout-equivalent budgets $m \in \{3, 5, 10\}$, the top-1 prediction from each of the first m offsets is concatenated in offset order, decoded as a single token sequence, and embedded once with MiniLM. Predictions are not fed autoregressively between offsets.