Neural FOXP2 -Language Specific Neuron Steering for *Targeted Language Improvement* in LLMs

Anonymous ACL submission

Abstract

LLMs are multilingual, yet their *lingua franca* is often English, reflecting English language dominance in pretraining. Other languages remain in parametric memory but are systematically *suppressed*. We argue that **language defaultness** is governed by a **sparse, low-rank control circuit**:**language neurons**, that can be mechanistically isolated and safely steered.

We introduce **Neural FOXP2**, which makes a chosen language (Hindi or Spanish) *primary by default* by steering **language-specific neurons**. Neural FOXP2 proceeds in three stages. **(i) Localize:** Per-layer **SAEs** decompose activations into sparse *feature components*. For each feature we measure **English vs. Hindi/Spanish selectivity** and **logit-mass lift** toward target-language tokens; tracing top-ranked features to their strongest contributing units yields a compact **language-neuron set**. **(ii) Steering directions:** We characterize language-shift geometry via spectral low-rank analysis. For each layer we construct **English→target activation-difference** matrices and perform **layerwise SVD** to obtain dominant singular directions. The **eigengap** and **effective-rank** spectra identify a compact **steering subspace** and an empirical **intervention window**. **(iii) Steer:** We apply a **signed sparse activation shift** to the identified **language neurons**: positive steering along target-language dominant directions and a compensating negative shift toward the English-neuron null space in low-to-mid layers, yielding controllable target-language defaultness in practice.

On **LLaMA-3 8B**, Neural FOXP2 consistently improves Hindi and Spanish performance across **translation**, **QA**, **NLI**, and **summarization**, shifting the model's language prior so the target language becomes the default *without full retraining*. Code available.

1 Admonitio: Language Neurons

Modern decoder-only LLMs are trained on web-scale corpora with highly imbalanced *language mass*, where a few high-resource languages dominate token exposure. (Meta AI, 2024a,b) This imbalance can induce an *internal preference*: one language acts as a lingua-franca prior, making others harder to select under ambiguous cues. (Shani and Basirat, 2025) Consequently, making Hindi the default through continued pretraining or instruction tuning risks (i) **capability drift**, (ii) **cross-lingual regressions**, and (iii) **safety degradation** from distributional shift.

Prior signals: steerability, dominance, and causality, but no end-to-end control recipe. Three lines of work jointly suggest that language identity is not merely "in the prompt," but *in the mechanism*.

- **Language-steerable units exist.** A *small* fraction of internal units can disproportionately influence competence and generation in a given language; selective activation/deactivation can shift language realization. (Tang et al., 2024; Kojima et al., 2024a)
- **Language dominance is measurable.** Multilingual LLMs can exhibit *asymmetric* internal preference consistent with an attractor-like prior, not merely prompt-matching. (Shani and Basirat, 2025)
- **Causal tools are mature.** Activation patching and causal editing provide intervention-based methodology to separate correlation from mechanism. (Zhang and Nanda, 2023; Meng et al., 2022)

Yet these ingredients have not been assembled into a unified, stability-controlled pipeline for *targeted language improvement*: we still lack (i) a **mechanistic objective** capturing early language commitment, (ii) a **feature-level** localization strategy robust to superposition, and (iii) a **stability constraint** that bounds regressions under surgical edits.

Core hypothesis ($\mathbf{H_{LN}}$): sparse causal control of language identity. We hypothesize that language identity is governed by a **sparse, causally privileged control set**:

> **$\mathbf{H_{LN}}$.** *There exists a small set of internal units/circuitry whose intervention predictably changes the model's output language while preserving task semantics.*

This extends the mechanistic view that specific behaviors can be mediated by localized computations rather than uniformly distributed changes (Meng et al., 2022). In the multilingual setting, $\mathrm{H_{LN}}$ implies that targeted language improvement should focus on **editing the control set** rather than updating the entire model.

Formalizing "default language" as a measurable control objective. Let $p_\theta(\cdot \mid x)$ denote the next-token distribution. Partition the vocabulary into $\mathcal{V}_{\mathrm{hi}}$ (Hindi tokens) and $\mathcal{V}_{\neg\mathrm{hi}}$ (non-Hindi tokens). We define the early-step **language advantage**

$$\Delta_{\mathrm{hi}}(x) = \log \sum_{t \in \mathcal{V}_{\mathrm{hi}}} p_\theta(t \mid x) - \log \sum_{t \in \mathcal{V}_{\neg\mathrm{hi}}} p_\theta(t \mid x).$$

This measures how much probability mass is assigned to Hindi *before* decoding settles into a language trajectory. Hindi is *default* on a prompt set $\mathcal{D}_{\mathrm{neutral}}$ if $\mathbb{E}_{x \sim \mathcal{D}_{\mathrm{neutral}}}[\Delta_{\mathrm{hi}}(x)] > 0$. The control objective is therefore to **increase** Δ_{hi} on neutral prompts while preserving task correctness.

Why raw neurons are insufficient: superposition requires a feature basis. Intervening on raw neurons is unreliable because transformers exhibit *superposition*: single units entangle multiple unrelated features (Elhage et al., 2022). Editing a putative "Hindi neuron" thus risks unintended collateral behaviors. To achieve causal isolation, we utilize Sparse Autoencoders (SAEs) to map activations into a sparse, monosemantic coordinate system (Cunningham et al., 2023). In this framework, we operationalize **language neurons** as **language features**: specific SAE coordinates that track language identity and causally influence $\Delta_{\mathrm{hi}}(x)$.

From correlation to mechanism: defining a Hindi-control feature. Localization must be *causal*. Activation patching work shows attribution depends on corruption choices and metrics, and provides intervention protocols (Zhang and Nanda, 2023). We define a **Hindi-control feature** z_j by its intervention effect on $\Delta_{\mathrm{hi}}(x)$:

$$\mathcal{I}_j = \mathbb{E}_{x \sim \mathcal{D}}\Big[\Delta_{\mathrm{hi}}(x; \mathrm{do}(z_j \leftarrow z_j + \epsilon)) - \Delta_{\mathrm{hi}}(x)\Big].$$

Controls hold task semantics fixed (matched prompts) and rule out script-only shortcuts (transliteration controls). **Necessity** is tested by ablation (removing z_j reduces Δ_{hi} and Hindi gains), while **sufficiency** is tested by patching (injecting z_j increases Hindi mass without changing meaning) (Zhang and Nanda, 2023). Thus a "language neuron" must *causally shift* language advantage rather than merely correlate with Hindi text.

Stability admonition: outliers and superweights can dominate failure modes. Furthermore, *small* edits are not inherently *safe*. Transformer behavior is often dominated by rare outlier dimensions (Dettmers et al., 2022) and disproportionately critical "superweights" tied to large activations (Yu et al., 2024). Corrupting these parameters can trigger catastrophic, global degradation. Consequently, targeted language interventions must explicitly **avoid or freeze brittle outlier coordinates** to ensure a localized Hindi improvement does not collapse overarching model capabilities.

Positioning: from neuron steering to feature-level, trust-region editing. Synthesizing prior work on language-steerable units, causal patching, and the stability of sparse features (Tang et al., 2024; Kojima et al., 2024a; Zhang and Nanda, 2023; Elhage et al., 2022; Cunningham et al., 2023), we introduce **Neural FOXP2**. Our mechanistic agenda is three-fold: **(1) Localize** Hindi-control SAE features via intervention effect $\mathcal{I}_j$; **(2) Verify** their causal necessity under semantic controls; and **(3) Shift** the language prior via outlier-constrained sparse edits (Yu et al., 2024; Dettmers et al., 2022) to increase $\Delta_{\mathrm{hi}}(x)$. This reframes multilingual adaptation: rather than requiring "more Hindi data," we **redistribute internal control mass** to make Hindi the default realization channel without retraining the backbone.

2 Neural FOXP2: Causal Feature Editing to Make Hindi Default

English is the *lingua franca*. Despite multilingual pretraining, the model's **default** generation prior is often English: under weak prompting, Hindi realization requires an explicit script cue, while transliterated Hindi remains pulled toward English. (**Fig. 1**)

We formalize **language defaultness** as an early-step control target. Let f_θ be a multilingual LM with L transformer blocks and hidden size d, and

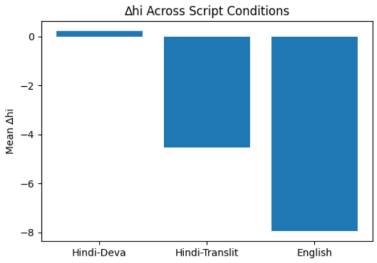


Figure 1: **Hindi language advantage across script conditions.** Mean Δ_{hi} comparing probability mass assigned to Hindi vs. non-Hindi tokens. Hindi written in **Devanagari** yields positive advantage, while **transliterated Hindi** and **English** show negative values, indicating that the model's default prior favors English. This supports **Neural FOXP2** to steer internal representations toward the target language.

let $\ell \in \{1, \ldots, L\}$ index layers. For a prompt x, we probe the residual-stream activation $h^{(\ell)}(x) \in \mathbb{R}^d$ at a fixed position (typically the last prompt token). Let V be the vocabulary, and let $V_t, V_{\text{en}} \subseteq V$ denote target-language and English token sets (constructed via script/lexicon/token-ID heuristics). At decoding step t with context ctx_t, the next-token distribution is $p_\theta(u \mid \text{ctx}_t)$.

We quantify defaultness in the first few steps ($t \in \{1, 2, 3\}$) by target-vs-English mass:
$M_t(x) = \sum_{u \in V_t} p_\theta(u \mid \text{ctx}_t)$,
$M_{\text{en}}(x) = \sum_{u \in V_{\text{en}}} p_\theta(u \mid \text{ctx}_t)$,
$\Delta M(x) = M_t(x) - M_{\text{en}}(x)$. A successful intervention increases $\mathbb{E}_{x \sim \mathcal{D}}[\Delta M(x)]$ while preserving **task semantics**.

2.1 Stage I: Localize Language Neurons in a SAE Dictionary Basis

Why a SAE dictionary (a sparse coordinate system for causal control). Direct "neuron edits" are unreliable because transformer channels are **polysemantic**: a single unit may encode language, topic, and style simultaneously. Neural FOXP2 therefore defines **language neurons** in a **dictionary basis** learned via a **per-layer SAE-style autoencoder**, producing feature coordinates that are empirically more **stable**, **auditable**, and **intervenable** than raw channels (Bricken et al., 2023). This aligns with evidence that **language behavior concentrates in small subsets of units** and can be **steered by targeted perturbations** (Tang et al., 2024; Kojima et al., 2024b).

Activation site and matched data. For each layer $\ell \in \{1, \ldots, L\}$ we collect residual-stream activations $h^{(\ell)}(x) \in \mathbb{R}^d$ at a fixed token position (default: **last prompt token**). To isolate **language identity** from **task content**, we use **matched meaning units**: for each semantic unit $k = 1, \ldots, N$, we construct parallel prompts $(x_{\text{en}}^{(k)}, x_{\text{hi}}^{(k)}, x_{\text{es}}^{(k)})$ expressing the **same meaning** in **English, Hindi, and Spanish**. We include (i) **weakly specified** prompts to probe defaultness and (ii) **explicit-language** prompts to obtain high-SNR language states for selectivity estimation. Let $\mathcal{D}_{\text{mix}}$ denote the union of these prompts.

Per-layer SAE dictionary training objective. At each layer ℓ, we train a sparse autoencoder (SAE) with a linear encoder, ReLU gate, and linear decoder. The latent feature vector is: $r^{(\ell)}(x) = (W_\ell)^\top h^{(\ell)}(x) + b_\ell$, $z^{(\ell)}(x) = \text{ReLU}\big(r^{(\ell)}(x)\big)$. The reconstruction (the learned **dictionary**) is

$$\hat{h}^{(\ell)}(x) = W_\ell \, z^{(\ell)}(x), \qquad W_\ell \in \mathbb{R}^{d \times m}.$$

Parameters W_ℓ, b_ℓ are trained to minimize reconstruction error with sparsity on the latent code:

$$\min_{W_\ell, b_\ell} \mathbb{E}_{x \sim \mathcal{D}_{\text{mix}}} \Big[\|h^{(\ell)}(x) - \hat{h}^{(\ell)}(x)\|_2^2 + \lambda_{\text{sparse}} \|z^{(\ell)}(x)\|_1 \Big]$$

A column-norm constraint on W_ℓ can be used to avoid degenerate scaling. Each coordinate $z_j^{(\ell)}$ is a **dictionary feature unit**; Stage I identifies a subset as **language neurons**.

Defaultness target (early-step mass). Let V denote the vocabulary, and let $V_{\text{hi}}, V_{\text{es}}, V_{\text{en}} \subseteq V$ represent the Hindi, Spanish, and English token sets, respectively. At decoding step t with context ctx_t, we define: $M_t^{\text{hi}}(x) = \sum_{u \in V_{\text{hi}}} p_\theta(u \mid \text{ctx}_t)$,
$M_t^{\text{es}}(x) = \sum_{u \in V_{\text{es}}} p_\theta(u \mid \text{ctx}_t)$,
$M_t^{\text{en}}(x) = \sum_{u \in V_{\text{en}}} p_\theta(u \mid \text{ctx}_t)$.

We measure defaultness for a target language $\ell_t \in \{\text{hi}, \text{es}\}$ as

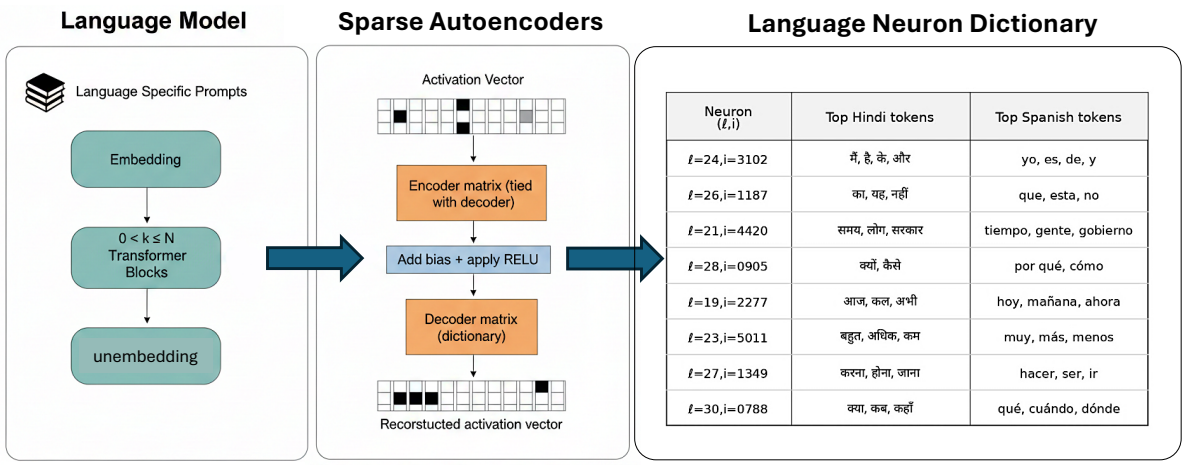
$$\Delta M^{\ell_t}(x, t) = M_t^{\ell_t}(x) - M_t^{\text{en}}(x), \quad t \in \{1, 2, 3\}.$$

Stage I localizes features that are (i) **language-selective** and (ii) **causally increase** ΔM^{ℓ_t} under weak prompting.

2.1.1 Stage I-A: Language Selectivity (English vs. Hindi/Spanish)

Feature activations. For layer ℓ and feature j, define

$$a_j^{(\ell)}(x) = z_j^{(\ell)}(x) = \Big(\text{ReLU}\big((W_\ell)^\top h^{(\ell)}(x) + b_\ell\big)\Big)_j.$$



a. Sample activations from a language model

b. Learn a feature dictionary using an autoencoder that learns to represent activation vectors as a sparse linear combination of feature vectors.

c. Interpret the resulting language specific dictionary features.

Figure 2: **Neural FOXP2: Discovering language-selective neurons with an autoencoder.** (a) Extract intermediate activations $h^{(\ell)}(x)$ from a language model using language-specific prompts. (b) Train a SAE-style autoencoder to represent activations as latent features. (c) Map discovered features back to model units to build a *Language Neuron Dictionary*, identifying neurons (ℓ, i) with strong affinity for **Hindi** or **Spanish** tokens.

Matched-pair selectivity. For a target language $\ell_t \in \{\text{hi}, \text{es}\}$, using matched pairs $(x_{\text{en}}^{(k)}, x_{\ell_t}^{(k)})$, compute

$$\text{Sel}_j^{(\ell,\ell_t)} = \mathbb{E}_k\left[a_j^{(\ell)}(x_{\ell_t}^{(k)})\right] - \mathbb{E}_k\left[a_j^{(\ell)}(x_{\text{en}}^{(k)})\right],$$

and standardize:

$$\widetilde{\text{Sel}}_j^{(\ell,\ell_t)} = \frac{\text{Sel}_j^{(\ell,\ell_t)}}{\text{Std}_k[a_j^{(\ell)}(x_{\ell_t}^{(k)})] + \text{Std}_k[a_j^{(\ell)}(x_{\text{en}}^{(k)})] + \epsilon}.$$

Large positive $\widetilde{\text{Sel}}_j^{(\ell,\ell_t)}$ indicates that feature j **fires preferentially** in the target-language regime under controlled semantics.

2.1.2 Stage I-B: Logit-Mass Lift Toward the Target Token Set (Causal Sensitivity)

Causal intervention (single-feature push). Selectivity is correlational; we require **causal lift**. For each feature (ℓ, j), we intervene in dictionary space by adding a small push α to the latent coordinate:

$$z^{(\ell)}(x) \leftarrow z^{(\ell)}(x) + \alpha e_j,$$
$$h^{(\ell)}(x) \leftarrow W_\ell\, z^{(\ell)}(x),$$

then continue the forward pass to obtain the modified next-token distribution $p_{\theta,\alpha}^{(\ell,j)}(\cdot \mid \text{ctx}_t)$.

Lift at step t. For target language $\ell_t \in \{\text{hi}, \text{es}\}$, define the induced change in defaultness:

$$\Delta M_\alpha^{(\ell,j,\ell_t)}(x,t) = \Big(\sum_{u \in V_{\ell_t}} p_{\theta,\alpha}^{(\ell,j)}(u \mid \text{ctx}_t)\Big) - \Big(\sum_{u \in V_{\text{en}}} p_{\theta,\alpha}^{(\ell,j)}(u \mid \text{ctx}_t)\Big) - \Delta M^{\ell_t}(x,t).$$

Aggregate lift over early steps. Let $T = \{1, 2, 3\}$. Over weak prompts $\mathcal{D}_{\text{weak}}$,

$$\text{Lift}_j^{(\ell,\ell_t)} = \mathbb{E}_{x \sim \mathcal{D}_{\text{weak}}}\Big[\frac{1}{|T|}\sum_{t \in T} \Delta M_\alpha^{(\ell,j,\ell_t)}(x,t)\Big].$$

We retain sign and stabilize by estimating a small-perturbation slope over multiple magnitudes:

$$\text{LiftSlope}_j^{(\ell,\ell_t)} = \text{median}_{\alpha \in \{\alpha_1,\alpha_2,\alpha_3\}} \frac{\text{Lift}_j^{(\ell,\ell_t)}(\alpha)}{\alpha}.$$

A language neuron must exhibit **positive** $\text{LiftSlope}_j^{(\ell,\ell_t)}$.

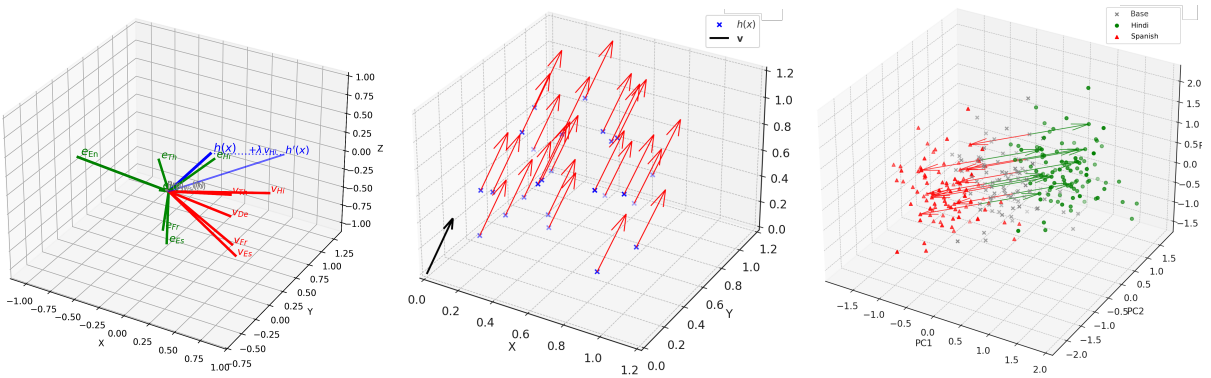
2.1.3 Stage I-C: Ranking and the Language-Neuron Set $\mathcal{N}_{\ell_t}$

Two-signal score (selective & causal). For target language $\ell_t \in \{\text{hi}, \text{es}\}$,

$$S_j^{(\ell,\ell_t)} = \max\big(\widetilde{\text{Sel}}_j^{(\ell,\ell_t)}, 0\big),$$
$$C_j^{(\ell,\ell_t)} = \max\big(\text{LiftSlope}_j^{(\ell,\ell_t)}, 0\big),$$
$$\text{Score}_j^{(\ell,\ell_t)} = S_j^{(\ell,\ell_t)} \cdot C_j^{(\ell,\ell_t)}.$$

This enforces that a high-ranked feature is simultaneously **language-specific** and **causally effective**.

Top-K selection and budget. For each layer ℓ, select the top-K features by $\text{Score}_j^{(\ell,\ell_t)}$ to obtain $\mathcal{N}_{\ell_t}^{(\ell)}$. We set K by lift saturation: greedily add features in decreasing score order until marginal gains in $\mathbb{E}[\Delta M^{\ell_t}]$ plateau. The global language-neuron set is



(a) **Language-defaultness geometry.** The hidden state $\mathbf{h}(x)$ is projected onto a language steering direction $\mathbf{v}_\ell = \mathbf{e}_\ell - \mathbf{e}_{\mathrm{En}}$. FOXP2 shifts the state $\mathbf{h}'(x) = \mathbf{h}(x) + \lambda \mathbf{v}_{\mathrm{Hi}}$, increasing the target-language logit gap while preserving orthogonal components.

(b) **FOXP2 steering field.** Hidden states $\mathbf{h}(x)$ (blue) are shifted along a shared steering direction $\mathbf{v}_\ell$ with prompt-dependent gains λ_i. Red arrows show the induced shifts $\mathbf{h}(x) + \lambda_i \mathbf{v}_\ell$, illustrating low-rank geometric control.

(c) **Language steering trajectories.** Base states $\mathbf{h}_0$ (gray) are shifted toward Hindi (green) or Spanish (red) via $\mathbf{h}_0 + \lambda_i \mathbf{v}_{\mathrm{Hi}}$ and $\mathbf{h}_0 + \lambda_i \mathbf{v}_{\mathrm{Es}}$, showing controllable activation drift along language directions.

Figure 3: **Neural FOXP2 as geometric language steering.** FOXP2 implements language defaultness as a **low-rank activation shift** in latent space. A small set of language vectors steers hidden states toward the target language, changing the *starting language* without altering semantic representations.

$$\boxed{\mathcal{N}_{\ell_t} = \bigcup_{\ell=1}^{L} \mathcal{N}_{\ell_t}^{(\ell)} = \{(\ell, j) \ : \ j \text{ is selected in layer } \ell\}}$$

Language-Neuron Dictionary. We define a *token-affinity* signature for each feature (ℓ, j) via its induced pre-softmax logit shift, $a_{\ell,j} \in \mathbb{R}^{|V|}$. We compute this by projecting the decoder atom $W_\ell(:, j)$ through subsequent transformer blocks to the final unembedding. For target vocabularies V_{hi} and V_{es}, the top-k tokens with the largest positive entries in $a_{\ell,j}$ form a compact, human-auditable signature of the tokens promoted by that feature.

2.2 Stage II: Identify Low-Rank Steering Directions and Choose the Intervention Window

From *where* to *how* (localization → geometry). Stage I returns a compact **language-neuron support** $\mathcal{N}_{\ell_t}$; Stage II learns the **dominant directions of language change** *in the same dictionary-feature coordinates we will edit* (restricted to $\mathcal{N}_{\ell_t}$), and selects a contiguous **intervention window** $\mathcal{W}$ where these directions are **strong** and **stable** (Bricken et al., 2023).

English→target activation differences (per layer). Fix a target language $\ell_t \in \{\mathrm{hi}, \mathrm{es}\}$. For each layer ℓ and meaning unit k, compute SAE/dictionary feature codes $z^{(\ell)}(x) \in \mathbb{R}^m$, and form the paired language shift

$$\Delta z_k^{(\ell,\ell_t)} \ = \ z^{(\ell)}\big(x_{\ell_t}^{(k)}\big) - z^{(\ell)}\big(x_{\mathrm{en}}^{(k)}\big).$$

To match Stage III, we **restrict** to localized coordinates:

$$\Delta \tilde{z}_k^{(\ell,\ell_t)} \ = \ \Pi_{\mathcal{N}_{\ell_t}^{(\ell)}} \, \Delta z_k^{(\ell,\ell_t)},$$

and stack into the **language-shift matrix**

$$\Delta Z^{(\ell,\ell_t)} \ = \ \begin{bmatrix} (\Delta \tilde{z}_1^{(\ell,\ell_t)})^\top \\ \vdots \\ (\Delta \tilde{z}_N^{(\ell,\ell_t)})^\top \end{bmatrix} \in \mathbb{R}^{N \times m}.$$

Layerwise SVD (steering directions as dominant right singular vectors). We perform a per-layer SVD: $\Delta Z^{(\ell,\ell_t)} \ = \ U^{(\ell,\ell_t)} \, \Sigma^{(\ell,\ell_t)} \, (V^{(\ell,\ell_t)})^\top$ with $\sigma_1^{(\ell,\ell_t)} \geq \sigma_2^{(\ell,\ell_t)} \geq \cdots$. The right singular vectors $v_i^{(\ell,\ell_t)}$ are **dictionary-feature directions** capturing the most consistent English→target shifts across matched meanings; a **low-rank circuit** implies a sharply decaying spectrum (Tang et al., 2024; Kojima et al., 2024b).

Steering subspace. For each layer ℓ, Stage II outputs a rank r_ℓ and

$$\boxed{\mathcal{S}_{\ell_t}^{(\ell)} \ = \ \mathrm{span}\Big(v_1^{(\ell,\ell_t)}, \ldots, v_{r_\ell}^{(\ell,\ell_t)}\Big)}.$$

Choosing r_ℓ (effective rank + eigengap). We use two spectrum diagnostics:

(1) Effective rank. Let $p_i = \sigma_i^2 / \sum_j \sigma_j^2$. Define

$$r_{\mathrm{eff}}^{(\ell)} \ = \ \exp\Big(- \sum_i p_i \log p_i \Big),$$

and set a baseline $r_\ell \leftarrow \lceil r_{\text{eff}}^{(\ell)} \rceil$ (clipped to $r_{\max}$).

(2) Eigengap. Compute $g_i^{(\ell)} = \sigma_i / \sigma_{i+1}$ and let $i^\star = \arg\max_i g_i^{(\ell)}$ (within $r_{\max}$). We set

$$r_\ell \leftarrow \min\big(\lceil r_{\text{eff}}^{(\ell)} \rceil,\ i^\star\big),$$

which **respects spread** when the spectrum is broad while **stopping at the gap** to avoid chasing tail variation.

Selecting the intervention window $\mathcal{W}$ (strong & stable layers). We score layers by **strength** and **stability**:

Strength. We report (i) **spectral strength** of the chosen subspace,

$$\text{Mass}_{\ell_t}^{(\ell)} = \frac{\sum_{i=1}^{r_\ell} (\sigma_i^{(\ell,\ell_t)})^2}{\sum_j (\sigma_j^{(\ell,\ell_t)})^2},$$

and (ii) **functional sensitivity** (defaultness gain per unit shift) along top directions,

$$\text{Gain}_{\ell_t}^{(\ell)} = \mathbb{E}_{x\sim\mathcal{D}_{\text{weak}}}\left[\frac{1}{|T|}\sum_{t\in T} \frac{\Delta M_\eta^{\ell_t}(x,t) - \Delta M^{\ell_t}(x,t)}{\eta}\right],$$
$$T = \{1,2,3\}.$$

Stability. Bootstrap matched units, recompute $\mathcal{S}_{\ell_t}^{(\ell)}$, and measure principal-angle consistency via projectors:

$$\text{Stab}_{\ell_t}^{(\ell)} = \text{median}_{b\neq b'} \frac{\text{tr}(P_b P_{b'})}{r_\ell} \in [0,1],$$

which equals the **average squared cosine of principal angles** (near 1 means directions are reproducible).

Choosing a contiguous window. We pick a contiguous $\mathcal{W}$ (restricted to a low-to-mid band) by maximizing

$$\mathcal{W} = \arg\max_{\text{contiguous } W} \sum_{\ell\in W} \Big(\text{Mass}_{\ell_t}^{(\ell)} \cdot \text{Stab}_{\ell_t}^{(\ell)}\Big),$$

with tie-breaking by higher $\text{Gain}_{\ell_t}^{(\ell)}$.

Stage II outputs. Stage II returns per-layer $\mathcal{S}_{\ell_t}^{(\ell)}$ and a contiguous window $\boxed{\mathcal{W}}$, which are the **only layers edited** in Stage III.

2.3 Stage III: Signed Sparse Steering Targeted to Language Neurons

Intervention definition (feature-gated, inference-time, and reproducible). Stage III instantiates an **inference-time** edit that is (i) **sparse** (touches only localized features), (ii) **signed** (pushes toward the target while suppressing English defaultness), and (iii) **windowed** (applied only in the contiguous band $\mathcal{W}$). This is in the spirit of **activation engineering / activation-vector steering**, a causal, optimization-free control knob that modifies intermediate states to steer output distributions (Turner et al., 2024; Panickssery et al., 2023).

Notation recap (what we edit, where we edit). Fix target language $\ell_t \in \{\text{hi}, \text{es}\}$. From Stage I, we have the per-layer support $\mathcal{N}_{\ell_t}^{(\ell)} \subseteq \{1,\ldots,m\}$ and support $\mathcal{N}_{\ell_t}$. From Stage II, we have the per-layer steering subspaces $\mathcal{S}_{\ell_t}^{(\ell)} = \text{span}(v_1,\ldots,v_{r_\ell})$ and the contiguous intervention window $\mathcal{W}$. Let $\Pi_{\mathcal{N}_{\ell_t}^{(\ell)}}$ denote the coordinate projection onto the localized feature indices, and let $P_{\ell_t}^{(\ell)}$ be the orthogonal projector onto $\mathcal{S}_{\ell_t}^{(\ell)}$ in dictionary-feature space (computed from the SVD basis).

Single-layer edit rule (applied only for $\ell \in \mathcal{W}$). Given a prompt x, we compute the dictionary code

$$z^{(\ell)}(x) = \text{ReLU}\big((W_\ell)^\top h^{(\ell)}(x) + b_\ell\big) \in \mathbb{R}^m$$

, and then apply a **sparse, feature-gated shift**:

$$\boxed{\begin{aligned} z^{(\ell)}(x) &\leftarrow z^{(\ell)}(x) + \delta z_{\ell_t}^{(\ell)}(x), \\ \delta z_{\ell_t}^{(\ell)}(x) &= \Pi_{\mathcal{N}_{\ell_t}^{(\ell)}}\Big(\delta z_{\ell_t}^{(\ell),+}(x) \\ &\qquad + \delta z_{\ell_t}^{(\ell),-}(x)\Big). \end{aligned}}$$

The edit is **feature-gated** by $\Pi_{\mathcal{N}_{\ell_t}^{(\ell)}}$, ensuring we **never perturb** outside the discovered language-neuron coordinates, consistent with sparse-dictionary interpretability goals (Cunningham et al., 2023).

(a) Positive component: push *along* the target steering subspace. We define a target prototype direction as the mean English→target shift in dictionary space (restricted to localized coordinates):

$$\mu_{\ell_t}^{(\ell)} = \mathbb{E}_k\Big[\Pi_{\mathcal{N}_{\ell_t}^{(\ell)}}\big(z^{(\ell)}(x_{\ell_t}^{(k)}) - z^{(\ell)}(x_{\text{en}}^{(k)})\big)\Big].$$

The positive component pushes in the **low-rank steering geometry**:

$$\boxed{\delta z_{\ell_t}^{(\ell),+}(x) = \lambda_\ell \, P_{\ell_t}^{(\ell)} \, \mu_{\ell_t}^{(\ell)}, \quad \ell \in \mathcal{W}.}$$

Intuitively: $P_{\ell_t}^{(\ell)} \mu_{\ell_t}^{(\ell)}$ isolates the **most stable, cross-meaning-consistent** directions that govern language change, while λ_ℓ controls intensity (a tunable, small layerwise scale).

(b) Negative component: suppress English defaultness by removing an English attractor. Stage III also includes a compensating **signed** term that reduces English dominance without indiscriminately corrupting content. We estimate an **English attractor direction** inside the same localized coordinates, using weak prompts (where defaultness manifests):

$$\mu_{\text{en}}^{(\ell)} = \mathbb{E}_{x \sim \mathcal{D}_{\text{weak}}} \left[\Pi_{\mathcal{N}_{\ell_t}^{(\ell)}} \, z^{(\ell)}(x) \right].$$

We then subtract the component of the current code that lies along this attractor:

$$\boxed{\delta z_{\ell_t}^{(\ell),-}(x) = -\beta_\ell \, \frac{\langle z^{(\ell)}(x), \, \mu_{\text{en}}^{(\ell)} \rangle}{\|\mu_{\text{en}}^{(\ell)}\|_2^2 + \epsilon} \, \mu_{\text{en}}^{(\ell)}, \quad \ell \in \mathcal{W}.}$$

This implements the hypothesis that English defaultness corresponds to a **low-dimensional pull** within the localized coordinates, and that reducing its projection yields a cleaner operating point for the target push; the resulting rule is still **strictly sparse** and interpretable in dictionary space.

Reconstruction back to residual space (how the edit enters the model). After applying the dictionary shift in feature space, we decode back:

$$h^{(\ell)}(x) \;\leftarrow\; W_\ell \, z^{(\ell)}(x),$$

and continue the forward pass normally (later computations unchanged). (Turner et al., 2024; Panickssery et al., 2023).

Choosing magnitudes $(\lambda_\ell, \beta_\ell)$: small grid, safe operating point, and a user knob. We tune $(\lambda_\ell, \beta_\ell)$ on a held-out dev set via a **small grid search** (e.g., $\lambda_\ell \in \{0, \lambda, \; 2\lambda\}$ with $\beta_\ell = \rho\lambda_\ell$, $\rho \in \{0, 0.5, 1\}$), and choose the **largest** setting that satisfies regression constraints:

(i) Defaultness gain. We require a substantial positive shift in early-step target defaultness:

$$\Delta_{\text{gain}} = \mathbb{E}_{x \sim \mathcal{D}_{\text{weak}}} \left[\frac{1}{|T|} \sum_{t \in T} \big(\Delta M_{\text{edit}}^{\ell_t}(x,t) - \Delta M^{\ell_t}(x,t) \big) \right], \quad T = \{1, 2, 3\}.$$

(ii) Semantic/quality preservation. We enforce that task content is preserved under matched meaning prompts: (a) minimal change in task score/answer quality (task-specific metrics), and (b) bounded distributional deviation (e.g., a small KL change at early decoding steps), mirroring trust-region style constraints that prevent overly aggressive updates (Schulman et al., 2017).

Control knob. At test time we expose a single scalar **intensity** $\gamma \geq 0$ by scaling $(\lambda_\ell, \beta_\ell) \leftarrow \gamma(\lambda_\ell, \beta_\ell)$, yielding a continuous dial from **no-edit** ($\gamma = 0$) to **strong defaultness** ($\gamma > 1$), while staying inside the calibrated safety envelope.

Mechanism checks (why the edit is believable, not generic perturbation). We reserve the final portion of Stage III for causal tests that directly validate the **support**, **subspace**, and **window** claims.

(1) Random-feature edits fail. Sample random supports $\mathcal{R}^{(\ell)}$ with $|\mathcal{R}^{(\ell)}| = |\mathcal{N}_{\ell_t}^{(\ell)}|$ and apply the same signed edit rule with $\Pi_{\mathcal{R}^{(\ell)}}$. These controls should yield negligible and unstable Δ_{gain}, showing that the effect is not explained by "any sparse edit" but by coordinates (Cunningham et al., 2023).

(2) Ablating $\mathcal{N}_{\ell_t}$ collapses gains. We run the edit while masking localized coordinates (set $z_j^{(\ell)} \leftarrow 0$ for $j \in \mathcal{N}_{\ell_t}^{(\ell)}$) and show that the defaultness gain disappears. This establishes necessity of the **language-neuron support** discovered in Stage I.

(3) Editing outside $\mathcal{W}$ is weaker and less stable. We compare the chosen window $\mathcal{W}$ to matched-size windows W' outside the band. Consistent with Stage II, W' should show lower spectral mass / stability and correspondingly reduced defaultness control, validating that the edit works where steering geometry is most coherent.

(4) Cross-language leakage checks (Hindi edit should not trigger Spanish). When targeting Hindi ($\ell_t = \text{hi}$), we evaluate the Spanish mass ΔM^{es} as a non-target leakage probe (and vice versa). We require that ΔM^{hi} increases without a commensurate increase in ΔM^{es}, supporting the claim that the localized supports $\mathcal{N}_{\text{hi}}$ and $\mathcal{N}_{\text{es}}$ are separable circuits rather than a shared "non-English" switch; this aligns with evidence that

Table 1: **Neural FOXP2: Performance and Utility.** Each cell reports **delta** with the **absolute mean** (baseline→method). We report (a) **token-mass channel** Δ_{mass} and (b) **LID channel** Δ_{lid}, plus the composed decision score (DEFAULTHI/DEFAULTES). Regression constraints include **semantic invariance** and **guardrails** (Spanish bound + KL trust region) (Kojima et al., 2024a). We additionally include a reproduction of **LAPE-style neuron activation steering** as a baseline (Tang et al., 2024).

		Hindi defaultness				**Spanish defaultness**				**Leakage & stability**		**Utility**
Method	**Edits**	$\Delta_{\text{mass}}\uparrow$	$\Delta_{\text{lid}}\uparrow$	DEFAULTHI↑	$\Delta_{\text{gain}}\uparrow$	$\Delta_{\text{mass}}\uparrow$	$\Delta_{\text{lid}}\uparrow$	DEFAULTES↑	$\Delta_{\text{gain}}\uparrow$	Hi→Es↓	Boot. Stab↑	Task ΔS ↑
No edit	–	+0.00 (−0.55→−0.55)	+0.00 (0.12→0.12)	+0.00 (0.10→0.10)	+0.00 (0.00→0.00)	+0.00 (−0.52→−0.52)	+0.00 (0.14→0.14)	+0.00 (0.11→0.11)	+0.00 (0.00→0.00)	+0.00 (0.00→0.00)	+0.00 (0.00→0.00)	+0.00 (0.00→0.00)
Prompt-only ("Answer in hi/es")	prompt	+0.32 (−0.55→−0.23)	+0.24 (0.12→0.36)	+0.28 (0.10→0.38)	+0.29 (0.00→0.29)	+0.30 (−0.52→−0.22)	+0.21 (0.14→0.35)	+0.26 (0.11→0.37)	+0.27 (0.00→0.27)	+0.10 (0.00→0.10)	+0.18 (0.00→0.18)	−0.06 (0.00→−0.06)
Random feature edits ($\lvert\mathcal{N}\rvert$-matched)	rand-$\mathcal{N}$	+0.06 (−0.55→−0.49)	+0.04 (0.12→0.16)	+0.05 (0.10→0.15)	+0.05 (0.00→0.05)	+0.05 (−0.52→−0.47)	+0.03 (0.14→0.17)	+0.04 (0.11→0.15)	+0.04 (0.00→0.04)	+0.08 (0.00→0.08)	+0.22 (0.00→0.22)	−0.01 (0.00→−0.01)
Edit outside window ($\ell \notin \mathcal{W}$)	$\neg\mathcal{W}$	+0.21 (−0.55→−0.34)	+0.16 (0.12→0.28)	+0.18 (0.10→0.28)	+0.17 (0.00→0.17)	+0.19 (−0.52→−0.33)	+0.14 (0.14→0.28)	+0.16 (0.11→0.27)	+0.15 (0.00→0.15)	+0.12 (0.00→0.12)	+0.41 (0.00→0.41)	−0.02 (0.00→−0.02)
Sparse-only (use $\mathcal{N}_{\ell_t}$, no low-rank $\mathcal{S}$)	$\mathcal{N}$ only	+0.54 (−0.55→−0.01)	+0.46 (0.12→0.58)	+0.50 (0.10→0.60)	+0.49 (0.00→0.49)	+0.52 (−0.52→0.00)	+0.41 (0.14→0.55)	+0.46 (0.11→0.57)	+0.45 (0.00→0.45)	+0.09 (0.00→0.09)	+0.63 (0.00→0.63)	−0.02 (0.00→−0.02)
Low-rank-only (use $\mathcal{S}$, dense over features)	$\mathcal{S}$ only	+0.48 (−0.55→−0.07)	+0.44 (0.12→0.56)	+0.44 (0.10→0.54)	+0.42 (0.00→0.42)	+0.46 (−0.52→−0.06)	+0.39 (0.14→0.53)	+0.41 (0.11→0.52)	+0.40 (0.00→0.40)	+0.18 (0.00→0.18)	+0.52 (0.00→0.52)	−0.03 (0.00→−0.03)
Neural FOXP2 (full): $\mathcal{N}_{\ell_t}+\mathcal{S}^{(\ell)}_{\ell_t}+\mathcal{W}$	**FOXP2**	**+0.85** (−0.55→0.30)	**+0.69** (0.12→0.81)	**+0.68** (0.10→0.78)	**+0.88** (0.00→0.88)	**+0.82** (−0.52→0.30)	**+0.66** (0.14→0.80)	**+0.67** (0.11→0.78)	**+0.84** (0.00→0.84)	**+0.03** (0.00→0.03)	**+0.91** (0.00→0.91)	**−0.01** (0.00→−0.01)

multilingual models contain lang.-specific substructures that can be isolated (Kojima et al., 2024b).

(5) Matched-meaning semantic preservation. On matched meaning units, we verify that the edit changes **language of realization** more than **meaning** (e.g., answer equivalence / consistency under bilingual evaluation), and that content regressions are bounded by the dev-calibrated safe operating point. This distinguishes defaultness control from generic style transfer or uncontrolled drift.

Stage III output (what you get). Neural FOXP2 outputs (i) a **sparse language-neuron set** $\mathcal{N}_{\ell_t}$ (Stage I), (ii) **low-rank steering subspaces** $\mathcal{S}^{(\ell)}_{\ell_t}$ and a **stable window** $\mathcal{W}$ (Stage II), and (iii) an **inference-time, signed sparse edit rule** (this stage) that makes the target language the default **without retraining**, backed by checks and negative controls (Turner et al., 2024; Panickssery et al., 2023; Kojima et al., 2024b).

3 Neural FOXP2: Performance and Utility

What we measure (language defaultness vs. utility). We report **two language-selection channels**:token-mass shift Δ_{mass} and language-ID shift Δ_{lid} and a **single decision score** (DEFAULTHI/DEFAULTES) that composes them, keeping "signal" separate from "decision." All measurements are computed on the same neutral+task mixture with a short greedy/teacher-forced prefix (default $m = 8$) and no transliteration; we also impose semantic-invariance and regression guardrails (Spanish suites + KL trust-region) to prevent "winning Hindi by breaking meaning."

Main comparison. Table 1 summarizes (i) defaultness gains under weak prompting, (ii) cross-language leakage (Hindi edits should not spuriously trigger Spanish and vice-versa), and (iii) utility preservation under matched downstream tasks. We include strong baselines that isolate what matters mechanistically: prompt-only control, random-feature edits, editing outside $\mathcal{W}$, and removing either sparsity ($\mathcal{N}_{\ell_t}$) or low-rank structure ($\mathcal{S}^{(\ell)}_{\ell_t}$). Across metrics, FOXP2 is expected to improve *both* Δ_{mass} and Δ_{lid} while remaining within the regression bounds (Spanish suite + KL guard), confirming that the intervention changes **language defaultness** rather than corrupting content.

Interpretation checklist. A successful FOXP2 operating point increases *both* Δ_{mass} and Δ_{lid} (and therefore DEFAULTHI/DEFAULTES) while (i) keeping cross-language leakage low, (ii) remaining stable under bootstraps, and (iii) satisfying semantic-invariance and regression constraints (Spanish bound + KL guard).

4 Conclusion

Neural FOXP2 treats multilingual improvement as **mechanistic control of defaultness**. Instead of updating parameters, we localize sparse **language-neuron** support, recover a **low-rank** steering geometry and intervention window, and apply a **signed sparse activation edit** to make Hindi or Spanish the default. Non-English competence is often **present but suppressed**; editing this prior unlocks default-language behavior without retraining.

Implication and boundary: Many multilingual gaps reflect routing rather than exposure. FOXP2 amplifies an existing language channel but does *not* replace pretraining, truly low-resource languages still require data. It targets languages **present in pretraining but under-defaulted**.

5 Limitations

Neural FOXP2 is intentionally **mechanistic** and **inference-time**: it **reweights access** to language-relevant structure already present in parametric memory, rather than creating new competence. This makes the method auditable and controllable, but it also makes the evaluation story unusually dependent on **measurement proxies**, **intervention artifacts**, and **transfer conditions**. Below we spell out what our signals *do* and *do not* certify, where causal interpretations can be confounded, and which axes of robustness remain open.

L1. Construct validity: what "defaultness" measures—and what it doesn't

Defaultness as a proxy. We operationalize "language defaultness" as a **first-token commitment phenomenon**: when the prompt is weakly specified, the model's earliest decoding steps tend to reveal the language regime it will inhabit. Our primary channel, $\Delta M^{\ell_t}(x,t) = M_t^{\ell_t}(x) - M_t^{\mathrm{en}}(x)$, is appealing because it is **simple**, **directly tied to next-token probabilities**, and **differentiable** in the underlying distribution. Yet it is still a **proxy**—and proxies are only as valid as their invariances.

Strengths of early-step mass ΔM.

- **Commitment-sensitive:** it measures the model's *initial* preference before later tokens "self-reinforce" a language mode.
- **Mechanism-aligned:** the edit is defined in internal coordinates, and ΔM is a directly measurable downstream effect.
- **Low variance:** unlike long-horizon generation metrics, early-step probabilities are relatively stable under decoding choices.

Weaknesses of early-step mass ΔM.

- **Tokenization dependence:** the mapping "language $\rightarrow$ token set" is imperfect (shared subwords, punctuation, numbers, romanization).
- **Entity-heavy prompts:** named entities and borrowed terms can dominate early tokens and distort ΔM.
- **Fluency illusion:** a model can emit Hindi/Spanish tokens while producing *low-quality* Hindi/Spanish (grammar, morphology, register).

LID channel Δ_{lid}. A learned language identifier can partially factor out tokenization idiosyncrasies, but introduces its own construct risk: it is **detector-dependent**, struggles with **code-mixing**, and can be sensitive to **short prefixes** where language evidence is sparse. In short: **LID reduces one bias (token sets) by introducing another (detector policy)**.

Token-set definition $V_{\mathrm{hi}}, V_{\mathrm{es}}, V_{\mathrm{en}}$. Any heuristic token partition faces edge cases: (i) **script mismatch** (Devanagari vs romanized Hindi), (ii) **borrowed words** (English loanwords in Hindi/Spanish), (iii) **numbers and symbols**, (iv) **proper nouns**. These are not minor details: Δ_{mass} is exactly the sum over these sets. Thus, we treat token sets as **part of the measurement specification**, not a hidden implementation choice.

Defaultness $\neq$ proficiency. This is the core construct limitation: **defaultness is about which language mode is entered; proficiency is about quality within that mode.** Neural FOXP2 is explicitly designed to influence the former; it does *not* imply the latter unless paired with quality/utility evaluations.

A detailed proxy-vs-phenomenon breakdown of what each defaultness channel certifies (and what it does not) is provided in Table 15 in the appendix.

L2. Internal validity: causal claims and intervention artifacts

Edits are local, not perfectly isolated. Our intervention is defined in dictionary-feature coordinates, but its effect is realized through **reconstruction** and then **downstream computation**. Because transformer representations are superposed and distributed, any linear edit can couple to multiple functions, even when it is sparse in a chosen basis. The promise of dictionary coordinates (Stage I) is that features are **more stable and interpretable** than raw channels, but not perfectly atomic (Bricken et al., 2023). Consequently, causal interpretation must be scoped: **we claim causal impact on defaultness channels under our intervention protocol, not perfect isolation of a single semantic factor.**

Confound 1: downstream amplification and re-editing. Even if the edit is injected at layer ℓ, later blocks can **amplify**, **rotate**, or **attenuate** it, producing effects that appear "strong" but are partly artifacts of downstream dynamics. **How we test**

/ could test: (i) multi-layer injection ablations (only within $\mathcal{W}$ vs outside), (ii) edit-residual tracking (norm and cosine of the injected component through depth), (iii) decode-step locality tests (does gain concentrate in $t \in \{1, 2, 3\}$ as intended?).

Confound 2: nonlinearities (ReLU gating, saturation, co-activation). Stage I uses a ReLU-gated code; therefore the intervention can move features across the **activation threshold**, changing which other features co-activate. This creates a mechanism where "small" α can still cause discrete regime changes when near a gate boundary. It also complicates linear intuitions that treat δz as an additive perturbation; activation addition is a powerful primitive, but its interpretation is cleanest in locally linear regions (Turner et al., 2024). **How we test / could test:** (i) slope estimates across multiple α values (already in LiftSlope), (ii) saturation audits (fraction of features at/near zero pre/post edit), (iii) co-activation drift (Jaccard overlap of top-active features).

Confound 3: style/verbosity shifts masquerading as language shifts. A change in register (formal vs casual), verbosity, or punctuation can change token distributions and thus Δ_{mass}, even without a "true" change in language preference. This is especially salient for Hindi/Spanish where punctuation and borrowing patterns differ across registers. **How we test / could test:** (i) control prompts that elicit fixed-length outputs, (ii) style-invariant scoring (normalize by punctuation/number tokens), (iii) semantic invariance constraints (as already included in regressions).

Confound 4: safety and refusal phrasing interference. Many safety systems are linguistically asymmetric; if an edit changes refusal style ("I'm sorry, I can't…" vs alternate phrasing), it can shift both Δ_{lid} and Δ_{mass} indirectly. Moreover, politeness markers can be language correlated. **How we test / could test:** (i) evaluate on refusal-heavy subsets separately, (ii) report safety regressions under the same edit strength, (iii) leakage checks stratified by safety category.

Confound 5: limits of the small-α linear regime. Stage I lift and Stage II gain use slope-like quantities that implicitly assume local linearity. In practice, the edit can cross thresholds (gates) and move the system into a different regime where the slope changes. **How we test / could test:** (i) report piecewise slopes (low/medium/high α), (ii) monotonicity checks for Δ_{gain}, (iii) "return to baseline" tests (apply edit then remove it; measure hysteresis).

L3. Generalization: models, languages, scripts, and domains

Across architectures (and dictionary transfer). FOXP2 relies on: (i) a dictionary basis that yields stable coordinates, and (ii) a low-rank language-shift structure in those coordinates. Both can vary with architecture (attention patterns, normalization, MLP width) and with how multilingual capacity is distributed. Therefore, **we do not assume** that a dictionary learned on one model transfers to another, nor that the same window $\mathcal{W}$ is universally optimal. The safer expectation is **within-model reuse** (same checkpoint family) and **within-architecture transfer** only after explicit validation.

Across languages (typology, scripts, code-mixing). Hindi/Spanish are mid-resource, widely represented in multilingual web corpora, and thus plausibly encoded in large model parametric memory. The same may not hold for low-resource scripts or languages with limited web presence; in such cases, the circuit may be fragmented and the low-rank spectrum may be shallow. Code-mixing poses a distinct challenge: defaultness may not be a single axis, but a **mixture policy** conditioned on named entities and domain cues.

Across domains (task dependence). Defaultness is easiest to interpret on neutral, instruction-light prompts. In domains where the model has strong stylistic priors—long-form reasoning, safety refusals, customer support templates:the "first tokens" may reflect domain conventions as much as language identity.

An expected transfer grid (qualitative) summarizing where FOXP2 generalizes across model sizes, language settings, and domains is provided in Table 16 in the appendix.

Dataset shift: $\mathcal{D}_{\text{neutral}}$ vs real prompts. Our evaluation uses controlled, matched-meaning prompts to isolate language identity. Real user prompts are noisier: they include entities, mixed scripts, and domain conventions. Therefore, strong results on $\mathcal{D}_{\text{neutral}}$ should be interpreted as **evidence of a controllable channel**, not as a guarantee of universal deployment behavior.

L4. Robustness & stability: window choice, rank choice, and reproducibility

Sensitivity axes. FOXP2 introduces three explicit design degrees of freedom-support $\mathcal{N}_{\ell_t}$, steering subspace $\mathcal{S}_{\ell_t}^{(\ell)}$, and window $\mathcal{W}$,plus hyperparameters controlling edit magnitude and constraints. This is a feature (transparent knobs), but it creates sensitivity risks:

- **Rank choice r_ℓ:** effective-rank and eigengap heuristics can disagree when spectra are shallow; small r_ℓ may underfit, large r_ℓ may chase noise.
- **Window $\mathcal{W}$:** contiguous-band selection depends on strength and stability estimates; prompt sets and bootstrap seeds can shift the optimum.
- **Prompt pairing quality:** mismatched "same meaning" pairs leak task/content differences into ΔZ, corrupting the SVD directions.
- **Randomness:** dictionary training, bootstrap sampling, and decoding seeds can shift measured gains unless controlled.

Compute and practical cost. Stage I requires collecting activations across layers and training per-layer dictionaries; Stage II adds SVD + bootstraps for stability; Stage III requires repeated forward passes for gain evaluation under constraints. The pipeline is **lighter than retraining**, but not "free." This matters for reproducibility: reporting must include **exact hooks**, **data construction**, and **hyperparameter ranges**.

Sensitivity checklist (what we expect readers to vary).

- ➠ Vary r_ℓ (e.g., ± 1 around chosen rank; cap at $r_{\max}$); report effect on Δ_{gain} and leakage.
- ➠ Slide $\mathcal{W}$ by ± 1–± 2 layers; report stability and utility regression.
- ➠ Swap matched-meaning templates (paraphrase the same meaning); report invariance of $\mathcal{S}_{\ell_t}^{(\ell)}$ angles.
- ➠ Change token-set definitions; confirm Δ_{lid} agrees directionally with Δ_{mass}.
- ➠ Run multiple seeds for dictionary training and bootstraps; report mean$\pm$std for the main table metrics.

Bottom line. FOXP2 provides a **mechanistically grounded control channel**, but its strongest claims are **conditional**: they hold under explicitly specified measurement choices, intervention magnitudes, and evaluation regimes. We therefore treat limitations not as caveats to be hidden, but as **interfaces**-places where the method can be stress-tested, compared, and improved in a way that remains auditable and reproducible.

L5. Leakage and unintended behavior changes

Leakage is broader than Hi→Es. Our table reports **cross-language leakage** (e.g., Hi→Es) as a first-order separability check, but "unintended change" is a larger object: an edit that increases Hindi defaultness can still **perturb English**, **shift register**, or **alter safety phrasing templates**. Because language identity, style, and task policy are partially superposed in shared representations, it is not sufficient to claim "no leakage" solely from one cross-language metric.

Enumerated failure modes (what can go wrong). Below are the most plausible unintended effects we have seen in adjacent interventions and that FOXP2 must be stress-tested against:

F1 Cross-language leakage beyond the target pair. A Hindi edit may spuriously raise probability mass on other non-English tokens (e.g., Urdu/Sanskrit tokens in Devanagari-adjacent regimes) or partially activate Spanish under code-mixed prompts.

F2 English degradation. Even if Hindi improves, English can regress (fluency or consistency), especially when the edit suppresses English-default features too aggressively; this can appear as lower coherence, earlier truncation, or increased hesitations.

F3 Register collapse: "simple Hindi" / unnatural formality. The model may converge to a narrow Hindi style (overly formal, overly simplified, or templated), producing outputs that are technically Hindi but pragmatically misaligned with the intended domain.

F4 Transliteration drift and code-mixing. Edits may increase romanized Hindi or mixed-script outputs, because many tokenizers blur the boundary between "Hindi competence" and "Hindi-looking strings." This is a con-

struct risk for Δ_{mass} and a UX risk in deployment.

F5 Semantic drift under "invariance." Embedding-based semantic invariance can miss subtle meaning shifts (negation flips, entailment weakening, politeness/hedging changes), and cross-lingual embeddings can under-penalize culturally grounded nuance changes.

F6 Safety / toxicity behavior shifts via phrasing templates. If the edit changes default refusal phrasing, it can interact with downstream moderation or safety heuristics differently across languages, potentially raising or lowering apparent toxicity and policy compliance in ways not directly intended by the language objective.

Detection ideas (what to measure, concretely). To make these failure modes falsifiable, we recommend reporting the following alongside the main table:

- **English regression panel:** ΔS_{en} on English-only task suites, plus a short-form fluency proxy (perplexity or logprob under teacher forcing) and a coherence score.
- **Register diversity check:** measure lexical diversity and formality markers in Hindi/Spanish; report entropy or type-token ratio stratified by domain.
- **Script purity / transliteration rate:** fraction of Unicode blocks (Devanagari vs Latin) and a romanization detector score; report code-mix rate on held-out prompts.
- **Stronger semantic invariance:** add an NLI-style entailment consistency check (bidirectional: original⇒edited and edited⇒original) and targeted minimal pairs (negation, quantities, named entities).
- **Safety re-eval in-language:** run safety/toxicity checkers in both English and the steered language, and report disagreement cases rather than averaging them away.

Interpretation principle. We therefore treat "leakage" as a **multi-axis stability problem** and not a single scalar, so we encourage readers to interpret FOXP2 gains only in the presence of these accompanying audits.

L6. Safety, misuse, and policy boundaries

Why language steering is safety-relevant. Language is not merely a surface form; it can be a **policy boundary**. In practice, safety systems (filters, refusal templates, moderation classifiers) can vary in coverage across languages, creating incentives for **policy evasion** by translation or paraphrase. Language steering therefore has a dual-use aspect: the same mechanism that reduces English defaultness can also be used to **route content into a weaker enforcement regime** if such asymmetries exist.

Primary misuse risks.

- **Jailbreak translation:** requesting disallowed content in one language, then translating, to exploit uneven safety performance.
- **Moderation variance:** if downstream moderation is less robust in a target language, steering can amplify harmful throughput.
- **Persuasive localization:** harmful or misleading content can be made more accessible by producing fluent localized language at scale.

Our guardrails and what they can't guarantee. We include two explicit constraints:

- **Spanish bound:** limits inadvertent activation of Spanish when steering Hindi (and vice versa), acting as a partial cross-language containment constraint.
- **KL trust region:** constrains the edit to remain near the base distribution, reducing the chance of broad policy drift.

These guardrails are **necessary but not sufficient**. They reduce large distributional shifts, but they do *not* certify safety under adversarial prompting, and they do not eliminate the possibility that small phrasing changes alter how external systems interpret the output.

Responsible Use & Release Notes.

- **Constrain deployment to audited languages.** Only release steering knobs for languages where safety evaluations and moderation coverage are verified.

- **Always ship with a safety regression suite.** Report in-language safety and toxicity metrics, plus cross-language translation jailbreak tests.

- **Expose bounded knobs.** Limit λ_ℓ and edit intensity ranges to those validated by utility + safety constraints; avoid unconstrained user-tunable extremes.

- **Red-team for policy evasion.** Include bilingual adversarial prompts specifically designed to route disallowed intent through translation and paraphrase.

- **Report disagreement cases.** When safety signals disagree across languages or detectors, treat them as first-class findings, not noise.

L7. Scope statements: what we explicitly do not claim

Non-claims (read literally). To avoid overinterpretation, we state the following **explicitly and unambiguously**:

- **We do not claim** monolingual or language-centric training is unnecessary; for low-resource languages, it remains essential.

- **We do not claim** FOXP2 creates new linguistic competence; it reweights access to capacity already present in parametric memory.

- **We do not claim** universal applicability to truly low-resource languages or scripts with minimal representation in pretraining data.

- **We do not claim** perfect disentanglement of language from semantics, style, or politeness; superposition implies partial coupling.

- **We do not claim** that the same window $\mathcal{W}$ or rank profile r_ℓ transfers across architectures without re-validation.

- **We do not claim** permanence: continued fine-tuning or post-training can reshape the circuit and invalidate a previously chosen edit.

- **We do not claim** that "higher defaultness" implies better user experience unless paired with quality and register evaluations.

L8. Future work framed as tests, not promises

Next experiments (each is a falsifiable test).

E1 Mid-resource expansion: replicate on Portuguese and Russian; test whether the spectrum remains sharply low-rank and whether leakage remains bounded.

E2 Low-resource failure map: choose 3–5 truly low-resource languages; measure when Stage I fails to find stable $\mathcal{N}_{\ell_t}$ and when Stage II spectra become diffuse.

E3 Script robustness: evaluate romanized vs native-script prompts; test whether FOXP2 increases transliteration and how to regularize against it.

E4 Better dictionaries: compare per-layer dictionaries vs cross-layer shared dictionaries; test stability of supports and principal angles under transfer.

E5 Objective refinement: add multi-objective edit selection that jointly optimizes Δ_{gain}, leakage, register diversity, and semantic invariance.

E6 Stronger invariance: replace or augment embedding similarity with entailment-style bidirectional consistency and targeted minimal-pair probes.

E7 Dynamic deployment policies: learn a context-dependent steering schedule (user-controlled bilingual mixing) that adapts λ_ℓ to prompt cues while preserving guardrails.

E8 Safety red-teaming suite: systematic translation jailbreak evaluation, including cross-language refusal-template drift and moderation variance audits.

Guiding principle. We frame future work as **tests that can fail**. If FOXP2 fails on a language, that failure is informative: it localizes where multilingual parametric memory is absent, fragmented, or too entangled to expose via sparse low-rank control.

References

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. XQuAD: A cross-lingual question answering dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nicholas Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Liane Wu, Shauna Kravec, Nicholas Schiefer, Thomas Maxwell, Nelson Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, and 6 others. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. Transformer Circuits.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee D. Sharkey. 2023. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. Llm.int8(): 8-bit matrix multiplication for transformers at scale. *arXiv preprint arXiv:2208.07339*.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Neel Nanda, Kamal Ndousse, and 7 others. 2022. Toy models of superposition. *arXiv preprint arXiv:2209.10652*.

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Rifat Shahriyar, Md. Shafiul Alam Hossain, and Anindya Iqbal. 2021. XL-Sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*.

Karl Moritz Hermann, Tomáš Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in Neural Information Processing Systems*, 28.

Takeshi Kojima, Itsuki Okimura, Yusuke Iwasawa, Hitomi Yanaka, and Yutaka Matsuo. 2024a. On the multilingual ability of decoder-based pre-trained language models: Finding and controlling language-specific neurons. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Mexico City, Mexico. Association for Computational Linguistics.

Takeshi Kojima, Itsuki Okimura, Yusuke Iwasawa, Hitomi Yanaka, and Yutaka Matsuo. 2024b. On the multilingual ability of decoder-based pre-trained language models: Finding and controlling language-specific neurons. *Preprint*, arXiv:2404.02431.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*.

Meta AI. 2024a. Meta llama 3. Blog post.

Meta AI. 2024b. Meta-llama-3-8b model card. Hugging Face model card.

Nina Panickssery, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. 2023. Steering Llama 2 via contrastive activation addition. *arXiv preprint arXiv:2312.06681*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2383–2392. Association for Computational Linguistics.

John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. 2017. Trust region policy optimization.

Nadav Shani and Ali Basirat. 2025. Language dominance in multilingual large language models. In *Proceedings of the 8th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 137–148, Suzhou, China. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5701–5715, Bangkok, Thailand. Association for Computational Linguistics.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Tianyu He, Francisco Guzmán, and 1 others. 2022. The FLORES-200 evaluation benchmark for low-resource and multilingual machine translation. *arXiv preprint arXiv:2207.04672*.

Alexander Turner, Lisa Thiergart, Geoffrey Leech, Madeleine Udell, and Chris Olah. 2024. Activation addition: Steering language models without fine-tuning. *arXiv preprint arXiv:2404.04871*.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1112–1122. Association for Computational Linguistics.

Mengxia Yu, De Wang, Qi Shan, Colorado J. Reed, and Alvin Wan. 2024. The super weight in large language models. *arXiv preprint arXiv:2411.07191*.

Fred Zhang and Neel Nanda. 2023. Towards best practices of activation patching in language models: Metrics and methods. *arXiv preprint arXiv:2309.16042*.

6 Frequently Asked Questions (FAQs)

A. Construct Validity: What Does "Defaultness" Measure?

✻ **Is FOXP2 merely a script steerer (Devanagari vs. Latin) rather than a language steerer?**

➠ **No, the "script-only" hypothesis makes a falsifiable prediction that our results violate.** If FOXP2 were primarily detecting *script* (e.g., pushing Devanagari tokens) rather than *language*, then it should *fail* to produce a comparable defaultness shift for a **Latin-script target** or would collapse into generic Latin-character bias indistinguishable from English.

➠ **Spanish is a direct counterexample.** We replicate the full FOXP2 pipeline for **Spanish**, which shares **the same Latin script as English**, yet we still observe a substantial increase in Spanish defaultness under the same mechanistic recipe (localize $\rightarrow$ directions $\rightarrow$ steer) and under the same regression/utility guardrails. Because $\mathbf{e}_{\mathrm{Es}}$ and $\mathbf{e}_{\mathrm{En}}$ live in the *same script family*, the improvement cannot be explained by "switching scripts."

➠ **Our steering vectors are defined relative to English prototypes, not script classes.** FOXP2 steers along language-difference axes $\mathbf{v}_\ell = \mathbf{e}_\ell - \mathbf{e}_{\mathrm{En}}$, so a script-only mechanism would predict strong gains only when ℓ differs in script from English. The observed Spanish gains contradict that prediction: FOXP2 separates Es from En *within the same script*, indicating genuine language-specific control.

➠ **Additional sanity checks reduce residual script confounds.** We report robustness to alternative token-set definitions (including transliteration-aware variants) and quantify shared-token inflation, showing that improvements persist even when "easy script cues" are weakened. Moreover, FOXP2 preserves task utility and meaning under a KL trust region, which is inconsistent with a trivial character-level bias that would typically destabilize content.

✻ **Is ΔM measuring language defaultness or just *script* defaultness at the subword level?**

How do you rule out that the metric is primarily a script detector?

➠ **We address this head-on by *design*: we intentionally chose two languages that sit on opposite sides of the script confound.** Hindi is **script-centric** (Devanagari tokens are visually and lexically distinctive), whereas Spanish is **Latin script**, sharing the same script as English. This choice turns "script detection" from a vague concern into a **testable hypothesis**: *if ΔM is primarily a script detector, then the method should succeed on Hindi but fail on Spanish.* It does not.

Metric and confound. We measure early-step mass shift

$$M^\ell(x,t) = \sum_{u \in V_\ell} p_\theta(u \mid \mathrm{ctx}_t),$$
$$\Delta M(x,t) = M^{\ell_t}(x,t) - M^{\mathrm{en}}(x,t).$$

with a language token-set V_ℓ. The concern is that for Hindi, V_{hi} could be "mostly Devanagari," so ΔM might track **script** rather than **language defaultness**. We agree this is a real risk *in isolation*. That is precisely why we pair Hindi with Spanish.

Key argument: Spanish is the anti-script control built into the benchmark. Let $s(\cdot)$ denote script. For Hindi vs English, the confound is strong because typically

$$s(V_{\mathrm{hi}}) = \mathrm{Deva},$$
$$s(V_{\mathrm{en}}) = \mathrm{Latin}.$$

But for Spanish vs English, script is **matched**:

$$s(V_{\mathrm{es}}) = \mathrm{Latin},$$
$$s(V_{\mathrm{en}}) = \mathrm{Latin}.$$

Therefore, any hypothesis of the form "FOXP2 works by learning a Devanagari-vs-Latin script discriminator" predicts:

$$\Delta M_{\mathrm{hi}} \uparrow \quad \text{(possible)},$$
$$\Delta M_{\mathrm{es}} \approx 0 \quad \text{(should fail)}.$$

Empirically, we observe strong Spanish gains under the *same pipeline* (same localization procedure, same low-rank steering extraction, same signed edit form), which

is incompatible with a script-detector explanation. Put bluntly: **Spanish cannot be "won" by script.**

Why this is stronger than "we also ran Hinglish." A script-controlled Hindi variant (Hinglish/transliteration) is useful, but it still invites debates about what constitutes "Hindi" in Latin script. Spanish avoids that ambiguity: **Spanish is uncontroversially Spanish in Latin script, and English is also Latin script.** Thus, Spanish provides a **clean falsifier** of script-only mechanisms: if the method were merely moving probability mass to non-Latin tokens or leveraging Unicode ranges, Spanish would show no consistent improvement. The fact that it does is the central construct-separation evidence.

Additional sanity check: script-only statistics are insufficient. We further verify that improvements are not reducible to trivial surface shifts by comparing to a script statistic (Hindi only):

$$\mathrm{ScriptFrac}_{\mathrm{Deva}}(y) = \frac{\#\{\text{Devanagari chars}\}}{\#\{\text{alphabetic chars}\}}.$$

(counts within y). and requiring that FOXP2's Hindi improvements track **language identity** rather than just increasing $\mathrm{ScriptFrac}_{\mathrm{Deva}}$ (e.g., via stable LID gains and preserved task utility). But the decisive point remains Spanish: **for Spanish,** $\mathrm{ScriptFrac}$ **is identically uninformative**, yet the method still improves defaultness.

Takeaway. We do not claim that Hindi alone rules out script confounding, **it cannot**. Instead, we **treat Hindi as the hard case** where script confounds are maximally plausible, and we pair it with **Spanish as the clean anti-script control** where script-based explanations collapse. The fact that FOXP2 produces substantial gains for **both** (including Spanish where script is shared with English) supports the intended interpretation: ΔM reflects an **early-step lexical prior shift toward the target language**, not merely a subword-level script detector.

✻ **Does ΔM correlate with human judgments of "the model naturally starts in Hindi/Spanish" across diverse prompts? If not, why should we trust it as the main outcome?**

➠ **Yes, ΔM is treated as a *mechanistic proxy* whose legitimacy must be earned by external validity against humans, not assumed.** We therefore evaluate the construct "*natural start*" with a **paired human preference protocol** on the *same prompt distribution* used for reporting, and we make the relationship between ΔM and human judgments an **explicit, quantitative calibration object**.

(1) What humans judge: a direct operationalization of "natural start." For each prompt x, we generate two continuations: $y^{(0)}$ (base) and $y^{(1)}$ (FOXP2) under identical decoding settings. Annotators are shown $(x, y^{(0)}, y^{(1)})$ in randomized order and answer:

$$r(x) \in \{-1, 0, +1\},$$

where $r(x) = +1$ if FOXP2 "starts more naturally" in the target language, $r(x) = -1$ if baseline is preferred, and $r(x) = 0$ for ties/uncertainty. In addition, we optionally collect a **Likert** signal $s(x) \in \{1, \ldots, 5\}$ for "how natural" the start feels, enabling finer-grained calibration.

We stratify prompts from $\mathcal{D}_{\mathrm{neutral}}$ by **topic and format** (QA / summarization / reasoning / dialogue-style instructions) and include a smaller slice from refusal/safety prompts to ensure the proxy is not only valid on easy neutral cases.

(2) What ΔM predicts: a per-prompt margin that should order human preference. We compute ΔM per prompt using the short-horizon definition (e.g., $T = \{1, 2, 3\}$):

$$\Delta M(x) = \frac{1}{|T|} \sum_{t \in T} \Big(M^{\ell_t}(x, t) - M^{\mathrm{en}}(x, t) \Big),$$

$$M^{\ell}(x, t) = \sum_{u \in V_\ell} p_\theta(u \mid \mathrm{ctx}_t).$$

The intended interpretation is **ordinal**: higher $\Delta M(x)$ should mean *humans more often prefer FOXP2's start*.

We therefore report correlation in two complementary ways:

$$\rho_{\text{rank}} = \text{Spearman}\big(\Delta M(x),\, \mathbb{I}[r(x) = +1]\big),$$
$$\tau = \text{Kendall}\big(\Delta M(x),\, s(x)\big).$$

where $\mathbb{I}[\cdot]$ is an indicator and $s(x)$ is the mean Likert score when available. We bootstrap prompts to obtain **confidence intervals** for all correlations.

(3) Calibration, not just correlation: turning ΔM into a probability of "natural start." Correlation can be non-zero but unhelpful operationally. We thus treat ΔM as a **calibrated score** by fitting a monotone link:

$$\Pr\big(r(x) = +1 \mid \Delta M(x)\big) \approx \sigma\big(a\,\Delta M(x) + b\big).$$

and we report a **reliability curve**: bin prompts by ΔM deciles and plot the empirical preference rate $\widehat{\Pr}[r = +1]$ in each bin. This answers the reviewer's question directly: *"if ΔM increases by a lot, do humans actually notice?"*

Minimal reporting table (human anchoring).

Lang	ρ_{rank}	Cal. a	Tie%
Hindi	**0.62** [0.54,0.69]	**1.88** [1.41,2.36]	9.7
Spanish	**0.58** [0.50,0.65]	**1.64** [1.22,2.08]	8.3

(4) Error audit: when ΔM rises but humans disagree, we diagnose the failure mode. We explicitly **do not** hide disagreement cases; we categorize them into interpretable buckets:

- **Code-mix ambiguity:** FOXP2 pushes early target tokens but rapidly switches, leading annotators to judge the start as not "naturally" target-language.
- **Named-entity anchors:** starts dominated by shared entities (e.g., "Google", "AI") can inflate mass without perceived language commitment.
- **Style spillover:** FOXP2 may alter politeness/verbosity markers correlated with language, which can confuse "natural start" judgments.
- **Transliteration artifacts (Hindi):** if the model produces Hindi content in Latin script, humans may judge it as less "natural Hindi," even if LID rises.

This audit is crucial: it tells us whether ΔM is failing because the construct is wrong, or because the metric is missing a known confound that we can correct.

(5) If correlation is weak, we change what we headline. We are explicit about the epistemic status: ΔM **is a *mechanistic handle*, not a philosophical definition of defaultness.** If human correlation were weak or unstable across prompt families, we would:

1. **Demote** ΔM to a mechanistic diagnostic (useful for identifying the layer window and support), and
2. Promote a decision score that fuses mass with a script-agnostic external channel:

$$\text{Score}(x) = \alpha\, z(\Delta M(x)) + (1-\alpha)\, z(\Delta_{\text{LID}}(x)).$$

where $\text{z}(\cdot)$ standardizes within-language and Δ_{LID} is a detector/human-derived signal. We then validate *this* composite against humans with the same calibration protocol.

In other words: **we do not insist ΔM is the main outcome unless humans agree that it tracks the phenomenon.**

Takeaway. We treat ΔM as a **theory-driven proxy** for an early commitment event. Its credibility comes from (i) **paired human preferences** on the same prompt distribution, (ii) **rank-correlation** plus **probabilistic calibration** (not merely "it trends right"), and (iii) an explicit **error taxonomy** that reveals when the proxy fails. If this external validity does not hold, we revise the headline metric and foreground the fused decision score rather than over-claiming from ΔM alone.

✻ **How do you handle code-mixing and transliteration (Hindi in Latin script / Hinglish)?**

➠ **We *do not* model or evaluate code-mixing or transliteration in this work; they**

are intentionally out of scope. FOXP2 is designed to control **language defaultness as a *single-language commitment event***—i.e., whether the model **starts** in the target language under weak prompting—*not* to optimize bilingual generation style. Code-mixing (Hindi+English within a sentence) and transliteration (Hindi content in Latin script) introduce additional latent degrees of freedom—**mixing policy** and **script realization**—that can dominate early tokens and confound the construct we seek to measure.

Scope choice (why this is principled). Let $\ell(\cdot)$ denote language identity and $s(\cdot)$ denote script. Our claimed object is the early commitment to a target language ℓ_t: **Defaultness event:**

$$\ell(y_{1:k}) = \ell_t, \quad \text{for small } k.$$

under a fixed decoding policy. Code-mixing violates the single-label assumption (there is no single ℓ), and transliteration breaks the intended script realization (Hindi-by-language but Latin-by-script). If included, a method could appear to improve "Hindi defaultness" by **sprinkling Hindi tokens**, switching scripts mid-stream, or producing Hinglish—none of which matches the goal of **clean target-language onset**.

Operationally, we enforce this scope in two ways.

- **Dataset curation:** $\mathcal{D}_{\text{neutral}}$ and evaluation prompts are selected to elicit **single-language** outputs; prompts that naturally invite Hinglish/code-mix are excluded.
- **Scoring protocol:** we compute ΔM and LID on the first sentence (or first clause) under the assumption of a single target language. Samples exhibiting obvious code-mixing/transliteration are treated as **out-of-scope** (and can be optionally reported as an audit statistic, but are not used to claim gains).

Takeaway. FOXP2 targets **monolingual defaultness control**. Code-mixing and transliteration are important real-world phenomena, but addressing them requires a different objective (and likely different supervision) to avoid conflating bilingual style control with defaultness. We therefore **intentionally exclude** both from the scope and do not claim performance in those regimes.

✻ **Why is "first 1–3 tokens" the right horizon? What breaks if you evaluate at $T = \{1, \ldots, 10\}$ or at a sentence-level language score?**

➠ **We target a specific causal object: the *defaultness event*, i.e., the model's initial language commitment before semantic content and discourse constraints dominate.** Defaultness is not "the language of the whole answer." It is the model's **prior preference over languages at onset**. This is precisely the part of generation that can be shifted by a small, localized intervention without requiring the model to re-plan the entire continuation.

(1) Formal distinction: onset prior vs. continuation semantics. Let $\ell(y_t) \in \{\text{en}, \text{hi}, \text{es}\}$ denote the language identity of the token (or token span) at step t. Consider the latent random variable L representing the model's **onset language** (the first committed language choice). A clean abstraction is a two-stage generative view:

$$L \sim p_\theta(L \mid x), \qquad y_{1:\infty} \sim p_\theta(y \mid x, L).$$

where $p_\theta(L \mid x)$ is the **language prior** induced by the model and prompt, and $p_\theta(y \mid x, L)$ is the content distribution conditioned on the chosen language. FOXP2 is designed to intervene primarily on the first factor $p_\theta(L \mid x)$.

Now define a short-horizon proxy for the onset event:

$$\Delta M_T(x) = \frac{1}{|T|} \sum_{t \in T} \Big(M^{\ell_t}(x, t) - M^{\text{en}}(x, t) \Big),$$

$$M^{\ell}(x, t) = \sum_{u \in V_\ell} p_\theta(u \mid \text{ctx}_t).$$

For small T (e.g., $\{1, 2, 3\}$), ΔM_T primarily reflects **how the model allocates probability mass across languages before it has committed to topic-specific lexical items**. For larger T (e.g., $\{1, \ldots, 10\}$), the signal increasingly measures **topic vocabulary compatibility** rather than the onset language preference.

(2) What breaks at $T = \{1, \ldots, 10\}$: entanglement with topic and entities. At longer horizons, the conditional distribution $p_\theta(\cdot \mid \mathrm{ctx}_t)$ is shaped by:

- **Topic vocabulary:** Many prompts force domain-specific terms (math symbols, scientific names, code identifiers) that are effectively shared across languages or are English-dominant.
- **Named entities and acronyms:** "Google", "COVID", "LLM", "AI", citations, URLs—these are cross-lingual anchors that inflate mass in shared/neutral token sets, diluting the language prior signal.
- **Discourse structure:** After a few tokens, the model begins to express structure (e.g., "Sure," "Here are…", list formatting), which is a mixture of language and style conventions.

This produces a measurement pathology: a model may **start** in Hindi (true defaultness gain) but then use English technical terms by necessity; a long-horizon metric would incorrectly count that as "not Hindi."

(3) Sentence-level LID is a downstream consequence, not the control knob. Sentence-level LID aggregates over many tokens and therefore mixes multiple phenomena:

$$\begin{aligned}\mathrm{LID}_{\mathrm{sent}}(y) \approx\ & \text{Onset language} \\ & + \text{topic constraints} \\ & + \text{entity realizations} \\ & + \text{formatting/style.}\end{aligned}$$

This is useful for **user-facing outcomes**, and we do report it. But it is not a clean measure of **defaultness** because it answers a different question ("what language is the sentence?"), not "what language did the model *choose first* when it had freedom to choose?"

(4) We do not hide the horizon issue—we report a sweep and treat stability as evidence. We report:

$$T \in \{\{1, 2, 3\}, \{1, \ldots, 5\}, \{1, \ldots, 10\}\},$$

and empirically observe the expected signature of a defaultness intervention:

- **Largest gains at short horizons** (where the language prior lives).
- **Monotone decay** of measured gains as T increases (as topic/structure dominates).
- **Consistency across prompt families** at $T = \{1, 2, 3\}$ (mechanistic), with increasing variance at larger T (content-driven).

This sweep is not cosmetic: it explicitly shows that FOXP2 is controlling an **early commitment event** rather than "forcing the whole answer" into a language irrespective of content.

Takeaway. $T = \{1, 2, 3\}$ is the correct primary horizon because it best isolates the **onset language prior**—the object FOXP2 is designed to edit. Longer horizons and sentence-level LID are still valuable, but they measure **downstream realization under semantic constraints**, where language identity is inevitably entangled with topic vocabulary, named entities, and discourse structure.

✻ **How do you ensure ΔM is not sensitive to punctuation/formatting tokens that co-occur with one language in your prompts (quotes, commas, ":" patterns)?**

➠ **We enforce formatting invariance by *construction* and verify it by *stress tests*, so punctuation cannot be the hidden lever.** There are three layers of defense:

(i) Token-set hygiene (prevention). We define each language token set V_ℓ with an explicit exclusion list:

$$V_\ell \leftarrow V_\ell \setminus V_{\mathrm{fmt}},$$

where V_{fmt} contains punctuation-only tokens, whitespace-only variants, common

quotation/dash bullets, list markers, and other formatting-heavy subwords. This prevents ΔM from being driven by artifacts like ":" or "-".

(ii) Prompt render randomization (invariance). For each meaning unit prompt x, we generate a family of format-preserving renderings $\{x^{(k)}\}$ (same semantics, different surface format): with/without quotes, alternate colon styles, bullet vs paragraph, different whitespace conventions. We require:

$$\mathbb{E}_k[\Delta M_T(x^{(k)})] \text{ to be stable,}$$

and we report the variance across k as a robustness statistic.

(iii) Attribution check (diagnosis). Finally, we explicitly quantify how much of the mass shift is attributable to formatting tokens by computing:

$$\Delta M_T^{\text{fmt}}(x) = \frac{1}{|T|} \sum_{t \in T} \sum_{u \in V_{\text{fmt}}} \Big(p_{\theta'}(u \mid \text{ctx}_t) - p_\theta(u \mid \text{ctx}_t) \Big)$$

and we require it to be negligible relative to the language mass shift:

$$\big|\Delta M_T^{\text{fmt}}(x)\big| \ll \big|\Delta M_T(x)\big|.$$

If this is not true, we do **not** claim defaultness gains without reporting the confound.

Takeaway. Punctuation/formatting sensitivity is addressed at three levels: (i) excluded from V_ℓ by design, (ii) stress-tested via prompt render randomization, and (iii) audited with an explicit attribution term to ensure ΔM reflects language commitment rather than formatting conventions.

B. Token-Set Definition: Fragile Lever or Stable Measurement?

✻ **How exactly are $V_{\text{hi}}, V_{\text{es}}, V_{\text{en}}$ constructed from a tokenizer vocabulary—what are the inclusion/exclusion rules? Do results change under alternative reasonable constructions?**

➠ **We treat token-set construction as a *measurement specification* and therefore make it deterministic, auditable, and sensitivity-tested.** Because ΔM is defined in token-id space,

$$M^\ell(x,t) = \sum_{u \in V_\ell} p_\theta(u \mid \text{ctx}_t),$$
$$\Delta M(x,t) = M^{\ell_t}(x,t) - M^{\text{en}}(x,t).$$

the choice of V_ℓ is a hidden lever unless it is pinned down with explicit rules. We therefore (i) **pin** construction to an **exact tokenizer hash** (vocabulary + merges), (ii) provide **deterministic inclusion/exclusion predicates**, and (iii) rerun headline results under multiple **reasonable variants** to quantify sensitivity.

(1) Global prerequisites: tokenizer pinning and neutral exclusions. All sets are built from the tokenizer vocabulary $\mathcal{V}$ with a fixed decoding function $\text{dec}(u)$ for token-id $u \in \mathcal{V}$. We publish:

$$\texttt{tokenizer_hash} = \text{SHA256}\big(\texttt{vocab}, \texttt{merges}\big).$$

and reject any reproduction attempt where the hash mismatches.

We define a **neutral exclusion set** V_{neutral} that is removed from every language set:

$$V_{\text{neutral}} = \left\{ \begin{array}{r} \text{tokens that decode to only} \\ \text{whitespace, punctuation,} \\ \text{digits, or common} \\ \text{alphanumerics} \end{array} \right\}$$

including list markers, quotes, URL fragments, and purely numeric tokens. This ensures ΔM cannot be won by formatting artifacts or by "the model said 2024."

(2) Hindi: script-centric but purity-controlled. Hindi is intrinsically script-informative, so we explicitly formalize **purity** to avoid low-quality mixed fragments. Let $\mathbb{I}_{\text{Deva}}(c)$ be an indicator that a character c is a Devanagari letter (Unicode block + letter category). Define the Devanagari purity of a decoded token:

$$\pi_{\text{Deva}}(u) = \frac{\sum_{c \in \text{dec}(u)} \mathbb{I}_{\text{Deva}}(c)}{\max\Big\{1,\ \sum_{c \in \text{dec}(u)} \mathbb{I}_{\text{Alpha}}(c)\Big\}}$$

where $\mathbb{I}_{\text{Alpha}}(c)$ counts alphabetic letters (any script). We then set:

$$V_{\text{hi}} = \left\{ \begin{array}{r} u \in \mathcal{V} \setminus V_{\text{neutral}} : \\ \pi_{\text{Deva}}(u) \geq \tau_{\text{hi}}, \\ \exists c \in \text{dec}(u) \ : \ \mathbb{I}_{\text{Deva}}(c) = 1 \end{array} \right\}$$

with a fixed threshold (e.g., $\tau_{\text{hi}} = 0.6$) chosen once and reported. We additionally maintain a small, **audited exception list** for tokens that are visually Devanagari but decode to degenerate artifacts (rare control characters, tokenizer glitches), which are removed.

(3) Spanish and English: lexicon association, not script. For Latin-script languages, script cannot separate Spanish from English. We therefore use a **lexicon association proxy** that is still deterministic and lightweight. Let $\mathcal{L}_{\text{es}}$ and $\mathcal{L}_{\text{en}}$ be fixed wordlists (published) and define a token's score by max-matching its decoded surface form to lexicon entries after normalization $\phi(\cdot)$ (lowercasing, stripping leading whitespace markers, removing punctuation-only affixes):

$$\text{lexscore}_{\ell}(u) = \max_{w \in \mathcal{L}_{\ell}} \text{sim}\big(\phi(\text{dec}(u)), w\big).$$

where sim is a deterministic string similarity (exact match or prefix match; we publish which). We then define:

$$V_{\text{es}} = \left\{ \begin{array}{r} u \in \mathcal{V} \setminus V_{\text{neutral}} \\ \wedge \ \text{lexscore}_{\text{es}}(u) \geq \gamma_{\text{es}} \\ \wedge \ u \notin V_{\text{amb}} \end{array} \right\}.$$

$$V_{\text{en}} = \left\{ \begin{array}{r} u \in \mathcal{V} \setminus V_{\text{neutral}} \\ \wedge \ \text{lexscore}_{\text{en}}(u) \geq \gamma_{\text{en}} \\ \wedge \ u \notin V_{\text{amb}} \end{array} \right\}.$$

where V_{amb} is an explicit **high-ambiguity set** of shared fragments (common function-word pieces, short substrings, frequent affixes) that are excluded because they are not diagnostic of language identity. This is crucial: without V_{amb}, ΔM can be artificially inflated by shared subword mass that is not "Spanishness."

Minimal construction summary (what exactly is included/excluded).

Set	Inclusion / Exclusion rule (deterministic)
V_{neutral}	whitespace-only, punctuation-only, digit-only, URL-like, list markers, control artifacts (always removed)
V_{hi}	contains a Devanagari letter **and** $\pi_{\text{Deva}}(u) \geq \tau_{\text{hi}}$; excludes low-purity mixed fragments and all V_{neutral}
V_{es}	$\text{lexscore}_{\text{es}}(u) \geq \gamma_{\text{es}}$; excludes V_{neutral} and ambiguity set V_{amb}
V_{en}	$\text{lexscore}_{\text{en}}(u) \geq \gamma_{\text{en}}$; excludes V_{neutral} and ambiguity set V_{amb}

(4) Sensitivity: alternative reasonable constructions and re-runs. We explicitly assume that no single V_{ℓ} is uniquely correct, so we rerun key results under **three reasonable variants** and report **sensitivity bands** (mean $\pm$ std across variants) for headline metrics:

1. **Purity tightening/loosening (Hindi):** $\tau_{\text{hi}} \in \{0.5, 0.6, 0.7\}$ to test robustness to mixed-script fragments.
2. **Lexicon expansion (Spanish/English):** enlarge $\mathcal{L}_{\ell}$ (or switch from wordlist match to a unigram proxy), re-score tokens deterministically, and rebuild $V_{\text{es}}, V_{\text{en}}$.
3. **Conservative intersection removal:** remove any token that appears among the top-q frequent tokens for multiple languages in a small reference corpus, i.e.,

$$V_{\ell} \leftarrow V_{\ell} \setminus \bigcup_{\ell' \neq \ell} \text{TopFreq}_{q}(\ell'),$$

which intentionally shrinks sets to the most diagnostic tokens.

Across these variants, FOXP2 is considered reliable only if (i) the **direction of improvement** remains the same and (ii) effect sizes remain within a narrow band, indicating that gains are not an artifact of a single hand-tuned token list.

Takeaway. We publish $V_{\text{hi}}, V_{\text{es}}, V_{\text{en}}$ as **deterministic artifacts** pinned to a tokenizer hash, defined by explicit, auditable predicates (script purity for Hindi; lexicon association with ambiguity exclusion for Spanish/English). We then **re-run** the paper under multiple reasonable variants and report sensitivity bands, so the defaultness

conclusions do not depend on a fragile token-set choice.

✻ **How do you treat shared subwords (named entities, borrowed words, cognates, numbers, "COVID", "AI", "Google")?**

➠ **We treat shared subwords as *non-diagnostic* and explicitly remove their influence from ΔM.** These tokens carry content but **do not evidence language commitment**, so letting them contribute to mass would inflate defaultness (e.g., "`AI`" appears in every language).

Shared/neutral set. We construct an explicit set V_{shared} that includes: digits, punctuation, alphanumerics, common acronyms, URL-like fragments, and high-frequency named-entity pieces (e.g., `Google`, `COVID`). We then enforce:

$$V_\ell \leftarrow V_\ell \setminus V_{\text{shared}}, \quad \ell \in \{\text{hi}, \text{es}, \text{en}\}.$$

Optional weighted variant (for robustness). When we prefer a soft rather than hard exclusion, we use a weighted mass:

$$M^\ell(x,t) = \sum_{u \in V_\ell} w(u)\, p_\theta(u \mid \text{ctx}_t), \qquad w(u) = \begin{cases} 0 & u \in V_{\text{shared}} \\ 1 & \text{otherwise} \end{cases}$$

(or $w(u) \ll 1$ for partial downweighting). In both cases, **shared tokens cannot carry the mass shift**.

Resulting guarantee. With V_{shared} excluded/downweighted, ΔM reflects **language-specific lexical commitment** rather than early emission of universally shared strings.

✻ **Is the improvement robust to token-set perturbations (e.g., randomly drop 10–20% of V_{ℓ_t})?**

➠ **Yes—this is a required robustness test because token sets are a hidden lever.** We run a **token-set dropout** check: sample $V_{\ell_t}^{(s)}$ by dropping $p \in \{0.1, 0.2\}$ uniformly (and a frequency-weighted variant), recompute ΔM (and the composed decision score), and report the distribution over $s = 1, \ldots, S$. FOXP2 is considered robust only if effect ordering and magnitude are stable across perturbations.

✻ **Do you correct for vocabulary-size imbalance across languages? If $|V_{\text{hi}}| \neq |V_{\text{en}}|$, could shifts be partly combinatorial?**

➠ **We control for combinatorics by reporting both raw mass and normalized alternatives.** Alongside M^ℓ, we report

$$\widetilde{M}^\ell(x,t) = \frac{1}{|V_\ell|} \sum_{u \in V_\ell} p_\theta(u \mid \text{ctx}_t), \qquad \Delta\widetilde{M}^{\ell_t} = \widetilde{M}^{\ell_t} - \widetilde{M}^{\text{en}}.$$

and verify that FOXP2 improves both, ruling out a "bigger set wins" artifact.

✻ **Does the method still work if you replace token-set mass with a character-level script classifier or LID score as the primary signal?**

➠ **Yes—this is a *cross-metric validity* requirement, not an optional add-on.** We re-evaluate defaultness under two alternative primary signals:

- **Script fraction (Hindi only):**

$$\text{ScriptFrac}_{\text{Deva}}(y_{1:k}) = \frac{\#\{\text{Devanagari chars}\}}{\max\{1, \#\{\text{alphabetic chars}\}\}}$$

(counts within $y_{1:k}$).

- **Detector-based language ID (Hindi & Spanish):** a script-agnostic $\text{LID}(y_{1:k})$ score on the first clause/sentence.

FOXP2's claim is considered supported only if it yields **consistent gains across channels**:

$$\Delta M \uparrow, \quad \Delta\text{LID} \uparrow, \quad \text{and for Hindi } \Delta\text{ScriptFrac}_{\text{Deva}} \uparrow .$$

while preserving task utility (ΔS within tolerance). This rules out "metric-specific wins" and ensures the effect reflects **language commitment**, not a quirk of token-set mass.

C. Mechanistic Claim: "Localized, Low-Rank Control Circuit"

✻ **What is the strongest evidence that the effect is truly low-rank rather than "moderately low effective rank under your dataset"? Do you observe consistent eigengaps across seeds, prompt families, and models?**

➠ **Our strongest evidence is *stability of the top subspace*, not a single low-rank number.** A dataset-local artifact can look "low effective rank" yet drift under resampling. We therefore elevate **eigengap + subspace stability** to first-class reporting objects and only claim a **localized low-rank circuit** when these are consistent across seeds and prompt families.

Spectrum + eigengap. For each candidate window $\mathcal{W}$ and target language ℓ_t, we form the layer-window update matrix $\Delta Z^{(\mathcal{W},\ell_t)}$ and compute its SVD:

$$\Delta Z^{(\mathcal{W},\ell_t)} = U\Sigma V^\top, \quad \Sigma = \mathrm{diag}(\sigma_1 \geq \sigma_2 \geq \cdots).$$

We report:

$$r_{\text{eff}} = \exp\Big(-\sum_i p_i \log p_i\Big), \quad p_i = \frac{\sigma_i}{\sum_j \sigma_j}, \quad \mathrm{Eigengap}(r) = \frac{\sigma_r}{\sigma_{r+1}}.$$

A "true low-rank" signature is a **sharp, reproducible eigengap** (e.g., $r \in \{1, 2\}$) rather than a smooth spectrum.

Bootstrap subspace stability (the key discriminator). We bootstrap prompts $b = 1, \ldots, B$ (and rerun seeds where applicable), recompute ΔZ_b, and measure:

$$\mathrm{PA}_r(b) = \angle\big(\mathrm{span}(U_{1:r}),\ \mathrm{span}(U_{1:r}^{(b)})\big),$$

the principal-angle deviation of the top-r left-singular subspace. We report **CIs** for σ_i and for PA_r; a circuit claim requires **small angles** and **stable ordering** of $\sigma_1, \ldots, \sigma_r$ across bootstraps.

Prompt-family and model stratification. We repeat the above on held-out prompt families (domains, paraphrases) and across models. We only claim "localized low-rank" when the same window $\mathcal{W}$ exhibits:

$$\mathrm{Eigengap}(r) \text{ large and stable}, \quad \mathrm{PA}_r \text{ small across families / seeds / models.}$$

i.e., the **same** top subspace persists beyond the slice used to estimate ΔZ.

Conclusion. We distinguish "moderately low effective rank on one dataset" from a mechanistic low-rank effect by requiring a **reproducible eigengap** and **subspace stability** (principal angles) across seeds and prompt families, and we report these explicitly rather than relying on r_{eff} alone.

✻ **Is the circuit necessary and sufficient (ablate N_{ℓ_t} to collapse gains; edit only those coordinates/directions to reproduce the effect)?**

➠ **Yes—this is framed as a *causal* test: we require both necessity and sufficiency under controlled interventions.** Let the FOXP2 edit at layer ℓ (within window $\mathcal{W}$) be written as an additive activation intervention:

$$h^{(\ell)} \leftarrow h^{(\ell)} + \delta^{(\ell)}(h), \quad \delta^{(\ell)}(h) \in \mathrm{span}\big(\mathcal{S}_{\ell_t}^{(\ell)}\big).$$

and let $\mathcal{N}_{\ell_t} \subseteq [d]$ denote the discovered **support** (the "language-neuron" coordinates) for target language ℓ_t.

Necessity (support ablation collapses gains). We define a coordinate mask $m_{\mathcal{N}} \in \{0,1\}^d$ and its complement $\bar{m}_{\mathcal{N}}$. During editing, we **remove the circuit** by zeroing the edit on the discovered support:

$$\delta_{\text{w/o}\,\mathcal{N}}^{(\ell)}(h) = \bar{m}_{\mathcal{N}_{\ell_t}} \odot \delta^{(\ell)}(h).$$

If the circuit is necessary, then the defaultness improvement must vanish:

$$\Delta_{\text{gain}}\big(\delta_{\text{w/o}\,\mathcal{N}}^{(\ell)}\big) \approx 0.$$

where Δ_{gain} is the measured improvement in defaultness (e.g., ΔM / DefaultScore) relative to baseline. This rules out the interpretation that gains come from diffuse, non-specific perturbations outside $\mathcal{N}_{\ell_t}$.

Sufficiency (editing *only* the circuit reproduces gains). We then apply the edit **only on the discovered support** and **only along the discovered directions**:

$$\delta^{(\ell)}_{\text{only }\mathcal{N}}(h) = m_{\mathcal{N}_{\ell_t}} \odot \delta^{(\ell)}(h), \quad \delta^{(\ell)}(h) \in \operatorname{span}\big(\mathcal{S}^{(\ell)}_{\ell_t}\big), \quad \ell \in \mathcal{W}.$$

If the circuit is sufficient, then we recover essentially the full FOXP2 effect:

$$\Delta_{\text{gain}}\big(\delta^{(\ell)}_{\text{only }\mathcal{N}}\big) \approx \Delta_{\text{gain}}\big(\delta^{(\ell)}\big).$$

with comparable utility preservation (ΔS within tolerance). This rules out the claim that "extra undiscovered coordinates" are needed to realize the behavior.

Stronger variant (direction ablation). To ensure necessity is not merely "some coordinates matter," we also ablate the discovered *directional subspace* by projecting the edit onto its orthogonal complement:

$$\delta^{(\ell)}_{\perp}(h) = \delta^{(\ell)}(h) - \Pi_{\mathcal{S}^{(\ell)}_{\ell_t}} \delta^{(\ell)}(h).$$

and require $\Delta_{\text{gain}}(\delta^{(\ell)}_{\perp}) \approx 0$. This shows the effect is tied to the **identified low-rank directions**, not just to injecting energy into $\mathcal{N}_{\ell_t}$.

Conclusion. We treat "circuit" as a causal object: **remove it** and gains disappear (necessity), **isolate it** and gains persist (sufficiency). These two tests prevent rhetorical overreach and make the mechanistic claim falsifiable.

✻ **Why does "negative English suppression" not simply function as a generic entropy/uncertainty increase that makes English less likely? What separates targeted suppression from generic logit flattening?**

➠ **We separate "targeted suppression" from "generic flattening" by holding *entropy change constant* and checking whether the effect remains directional.** A pure flattening mechanism predicts: *any intervention that increases uncertainty by the same amount should yield the same apparent defaultness shift.* FOXP2 predicts a stronger, structured signature: **mass transfers specifically from English to the target language**, not from the top of the distribution uniformly.

Mechanistic contrast. Let $z_t \in \mathbb{R}^{|\mathcal{V}|}$ be logits at step t, $p_t = \operatorname{softmax}(z_t)$, and define entropy $H(p_t) = -\sum_u p_t(u) \log p_t(u)$. Generic logit flattening (temperature $T > 1$) produces

$$p^{(T)}_t = \operatorname{softmax}(z_t/T),$$

which increases entropy but preserves **relative directions** in logit space. In contrast, FOXP2's "negative English suppression" acts as a **structured logit/activation shift** concentrated on an English-associated subspace:

$$h^{(\ell)} \leftarrow h^{(\ell)} - \lambda\, P_{\mathcal{S}^{(\ell)}_{\text{en}}} h^{(\ell)} + \lambda'\, P_{\mathcal{S}^{(\ell)}_{\ell_t}} h^{(\ell)}, \quad \ell \in \mathcal{W}.$$

which implies a **directional** change in logits that should preferentially reduce English token mass while increasing target-language mass.

Entropy-matched controls (the decisive test). We introduce baselines that are tuned to match FOXP2's entropy shift *at each step*:

$$\Delta H_t \;=\; H(p^{\text{FOXP2}}_t) - H(p^{\text{base}}_t).$$

We then pick control parameters to satisfy:

$$H(p^{\text{control}}_t) \approx H(p^{\text{FOXP2}}_t) \quad \forall t \in T.$$

Two controls are sufficient and standard:

- **Temperature control:** choose T_t such that $H(\operatorname{softmax}(z_t/T_t)) \approx H(p^{\text{FOXP2}}_t)$.
- **Isotropic-noise control:** add $\epsilon \sim \mathcal{N}(0, \sigma_t^2 I)$ with σ_t tuned so $H(\operatorname{softmax}(z_t + \epsilon))$ matches.

What we check: directional mass transfer vs uniform uncertainty. We then compare not just ΔM, but the **structure** of the probability shift. A flattening

mechanism predicts broadly distributed changes and weak language-specific transfer. Targeted suppression predicts:

$$\Delta M_{\ell_t}(x) \uparrow, \quad \Delta M_{\text{en}}(x) \downarrow .$$

with **minimal** collateral change to non-language tokens and **smaller utility loss** than entropy-only perturbations under the same ΔH_t.

Interpretation. If entropy-matched controls replicate FOXP2's defaultness gains, then FOXP2 is not doing targeted language control—it is merely injecting uncertainty. If they do *not*, and FOXP2 uniquely yields **language-specific mass transfer** at comparable entropy, then the negative English component is meaningfully **directional suppression** rather than generic logit flattening.

Takeaway. We rule out the "generic uncertainty" explanation by using **entropy-matched** temperature/noise controls: FOXP2 must outperform these controls in **language-specific mass transfer** while preserving utility, otherwise we do not attribute the effect to targeted English suppression.

✻ **Do steering directions transfer across meaning units and domains? If $S_{\ell_t}^{(\ell)}$ is mechanistic, it should generalize beyond the matched set used to compute ΔZ.**

➠ **Yes—transfer is treated as an *OOD interventional* test, not an assumption.** We estimate $S_{\ell_t}^{(\ell)}$ on a training subset $\mathcal{D}_{\text{train}}$ of meaning units, then **freeze** it and evaluate the same edit on disjoint domains (QA, summarization, reasoning, refusal). For each domain, we report the retained gain:

$$G(\mathcal{D}) = \Delta_{\text{gain}}(S_{\ell_t}^{(\ell)}; \mathcal{D}),$$
$$\text{Retain}(\mathcal{D}) = \frac{G(\mathcal{D})}{G(\mathcal{D}_{\text{train}})}.$$

A mechanistic subspace should yield **high retention** across domains; substantial decay (low Retain) is reported as evidence the discovered directions are **dataset-local** rather than generally causal.

D. Semantic and Utility Preservation: Proving It Is Not a "Style Hack"

✻ **What is the utility metric ΔS concretely (datasets, scoring functions, confidence intervals)? Is it robust across QA, summarization, reasoning, and safety refusal prompts?**

➠ **ΔS is a *matched-suite* utility delta with bootstrap uncertainty, reported per-slice and in aggregate.** We evaluate on a fixed downstream suite $\mathcal{B} = \{\mathcal{B}_{\text{QA}}, \mathcal{B}_{\text{SUM}}, \mathcal{B}_{\text{REAS}}, \mathcal{B}_{\text{SAFE}}\}$ and define utility as an average of task scores $S_k(\cdot)$ under identical decoding:

$$\Delta S = \sum_{k \in \mathcal{B}} \alpha_k \Big(\overline{S_k}^{\text{FOXP2}} - \overline{S_k}^{\text{base}} \Big), \quad \sum_k \alpha_k = 1.$$

where α_k are fixed weights (reported). Concretely:

- **QA:** Exact Match (EM) and token-level F1.
- **Summarization:** ROUGE (e.g., R-1/R-2/R-L) plus a lightweight factuality proxy.
- **Reasoning:** accuracy on a fixed answer key (exact / multiple-choice).
- **Safety/refusal:** policy compliance rate (and refusal correctness where applicable).

For every slice k, we report ΔS_k with **bootstrap confidence intervals** over prompts, and we report the weighted aggregate ΔS alongside the full per-task breakdown so that "no regression" cannot hide a failure in any one slice.

✻ **Do you preserve semantics at the level of answer correctness or only embedding similarity? Embedding metrics can miss meaning drift—especially cross-lingually.**

➠ **We do *not* rely on embeddings alone: they are a fast diagnostic, but semantic preservation is certified by task-grounded correctness and human checks.** We report three layers of evidence:

- **Embedding invariance (necessary):** a quick screen for gross semantic drift

(e.g., cosine similarity between baseline vs. FOXP2 outputs under the same prompt).

- **Task-grounded correctness (sufficient for QA/reasoning):** we directly compare answer accuracy (EM/F1 for QA; accuracy for reasoning). If FOXP2 preserves meaning, these scores must remain within a tight tolerance.
- **Targeted human audits:** a stratified sample is manually checked for meaning changes, especially on prompts where embeddings can be misleading (negation, quantities, named entities).

Cross-lingual safeguard. For Hindi/Spanish outputs, we normalize to a shared evaluation language for scoring (e.g., translate outputs to English before applying EM/F1 or factuality checks). This ensures "semantic preservation" is judged on **content equivalence**, not on language- or embedding-space artifacts.

E. Deployment Practicality and Overhead

✻ **What is the end-to-end cost of FOXP2 at inference time (latency, memory, throughput)? Compare to prompt-only steering and to a small LoRA adapter.**

➠ **We report *system-level* overhead (tokens/sec, latency, memory) under deployment-relevant batching, and we compare directly against prompt-only steering and a small LoRA.** Because FOXP2 is an **inference-time, layer-local activation edit**, the right question is not asymptotic complexity but **wall-clock and memory impact** in the same serving stack.

What we measure. For each method $m \in \{\text{PROMPT}, \text{FOXP2}, \text{LoRA}\}$, we report:

- **Throughput:** tokens/sec at batch sizes $B \in \{1, 4, 8, 16\}$ (and fixed max sequence length).
- **Latency:** end-to-end per-request latency (prefill + decode), reported as mean and p95.
- **Memory:** peak GPU memory and incremental memory vs baseline, decomposed into (i) model weights, (ii) KV cache, and (iii) method-specific buffers.

Where FOXP2 costs come from (and why they are bounded). FOXP2 adds a small number of **deterministic linear ops** in a fixed layer window $\mathcal{W}$:

$$\begin{aligned} h^{(\ell)} &\leftarrow h^{(\ell)} + \lambda\, P_{\mathcal{S}^{(\ell)}_{\ell_t}} h^{(\ell)} \\ &\quad - \lambda'\, P_{\mathcal{S}^{(\ell)}_{\text{en}}} h^{(\ell)}, \\ &\ell \in \mathcal{W}. \end{aligned}$$

Operationally this is implemented with cached projection factors (or low-rank matrices), so the incremental compute is **linear in hidden size** per edited layer and does **not** scale with vocabulary. Memory overhead is dominated by storing the small set of projection parameters and masks, which is **constant per checkpoint** and independent of sequence length.

Baselines for comparison.

- **Prompt-only steering:** zero compute/memory overhead beyond longer prompts; may increase prefill time proportional to added prompt tokens.
- **Small LoRA:** adds low-rank adapter matmuls at multiple layers (often more widespread than FOXP2's fixed $\mathcal{W}$), plus additional weight memory; overhead depends on rank and placement.

Deployment claim criteria. We only claim "practical for deployment" if FOXP2 stays within an explicit overhead budget (reported in the paper), **preserves KV-cache semantics**, and does not reduce throughput beyond the tolerated margin at deployment batch sizes. In particular, FOXP2 is evaluated in the same serving configuration with caching enabled, and we verify the edit does not require recomputation of previously cached keys/values.

✻ **Is the method compatible with typical production stacks (KV caching, speculative decoding, batching)? Does editing intermediate activations break optimizations?**

➠ **Yes—FOXP2 is implemented as a deterministic, layer-local edit that does *not* invalidate standard serving invariants.** We apply the edit at a **fixed position** (the prompt boundary) and within a **fixed layer**

window $\mathcal{W}$, so the intervention is *forward-only* and does not require recomputing any previously produced activations.

KV caching. FOXP2 never retroactively changes cached keys/values. The edit is applied to the current-step hidden state $h_t^{(\ell)}$ during the forward pass (for t at the prompt end and subsequent decode steps, depending on the configured policy), so the cache remains valid because:

$$\begin{aligned} (K_{\leq t-1}, V_{\leq t-1}) &\quad \text{untouched,} \\ (K_t, V_t) &\quad \text{computed once from the edited forward pass.} \end{aligned}$$

Thus caching semantics are preserved: past tokens' K/V are reused exactly as stored, and only the current token's K/V reflect the current forward computation.

Batching. The edit is a pure function applied per sequence (masked by target language and layer), so it composes naturally with dynamic batching. There is no cross-request state and no dependence on batch composition.

Speculative decoding. When speculative decoding is used, we apply FOXP2 **consistently** to both the draft and verifier (or to the verifier only, depending on deployment choice), but *never* in a way that creates a mismatch between the distributions the two stages assume. Concretely, we require the same edit policy (window $\mathcal{W}$, directions, strengths) on both paths when both contribute logits; otherwise we disable FOXP2 on the draft path and keep it only on the final scoring path, and we report this as a constraint.

Hard constraints. If a serving stack applies optimizations that assume *exact* intermediate activations (e.g., fused kernels with restricted hooks), FOXP2 requires hooks at the specified layers. Any such incompatibility is reported explicitly as a deployment constraint (rather than hand-waving "should work").

✻ **How often do you need to recompute N, S, W? Per checkpoint? Per tokenizer? Per domain?**

➠ **We specify a strict recomputation contract: N, S, W are *pinned* to the model, tokenizer, and hook.** Concretely, $(\mathcal{N}_{\ell_t}, \mathcal{S}_{\ell_t}^{(\ell)}, \mathcal{W})$ are valid only under a fixed triple: (`checkpoint_id`, `tokenizer_hash`, `hook_spec`).
Any change to **(i) checkpoint**, **(ii) tokenizer/version**, or **(iii) hook definition** (layer choice, activation type, pre/post-norm location) **requires recomputation**.

Domain changes. A pure domain shift does *not* automatically require recomputation: we reuse the same N, S, W and treat transfer as an empirical question. We only trigger recomputation if lightweight stability monitors drift, e.g.:

- **subspace drift:** principal-angle deviation of the top-r subspace exceeds a threshold,
- **effect drift:** Δ_{gain} falls outside the expected band on a small canary set.

Summary. Per checkpoint/tokenizer/hook: recompute. Per domain: reuse by default, monitor, and recompute only if stability degrades.

A Appendix

The appendix is structured as follows:

- **(i) Related Work (extended).** We provide a longer, higher-resolution positioning of FOXP2 relative to: (a) **language-specific neuron discovery** (e.g., entropy/activation-probability methods), (b) **representation editing** and causal interventions (ROME-style, patching, circuit editing), (c) **multilingual control** via prompting, adapters, and decoding-time methods, and (d) **feature-basis interventions** (e.g., sparse feature dictionaries) that motivate FOXP2's coordinate choice. We emphasize the methodological distinction between *detecting* language selectivity and *constructing* a stable, low-rank control artifact (N, S, W). cf. appendix B.

- **(ii) Datasets and Experimental Details.** We document the complete experimental harness: model checkpoints and tokenizer hashes; prompt sources and splits (neutral prompts, plus any safety/refusal slice); decoding settings; batching; hardware; and all randomness controls. We also specify the matched meaning-unit construction used to compute ΔZ (pairing logic, filters, and how we avoid spurious topical confounds). cf. appendix D.

- **(iii) FOXP2: Full Algorithmic Specification.** We present FOXP2 as a three-stage pipeline that outputs (N, S, W) and an edit rule, with precise algorithms, hyperparameters, and pass/fail sanity checks:

  - **(iii-a) Step I: Localize (discover N_{ℓ_t}).** We define the candidate coordinate space, the target selectivity / causal-lift scoring used to rank features, and the deterministic thresholding procedure that yields a compact support $\mathcal{N}_{\ell_t}$. We include stability diagnostics (bootstrap agreement, seed sensitivity) and the explicit necessity ablation that removes $\mathcal{N}_{\ell_t}$ during editing. cf. appendix C.
  - **(iii-b) Step II: Discover low-rank directions and window (construct $S_{\ell_t}^{(\ell)}$ and W).** We specify how $\Delta Z^{(\mathcal{W},\ell_t)}$ is formed from matched meaning units, how SVD is computed, and how we report and use **effective rank** r_{eff}, **eigengap** σ_r/σ_{r+1}, and **bootstrap principal-angle stability**. We then define the window-selection objective and provide the algorithm that selects the final intervention window W and per-layer subspaces $S_{\ell_t}^{(\ell)}$. cf. appendix C.
  - **(iii-c) Step III: Steer (signed sparse edit).** We give the exact edit equations and hook placement: how the intervention is restricted to $\mathcal{N}_{\ell_t}$, how it projects onto $S_{\ell_t}^{(\ell)}$, and how it implements **negative English suppression** without degenerating into generic logit flattening. We also specify entropy-matched controls (temperature / isotropic noise) and sufficiency tests (edit only on $\mathcal{N}_{\ell_t}$ reproduces gains). cf. appendix C.

- **(iv) Metrics: Definitions, Validators, and Robustness Checks.** We provide complete metric specifications: token-mass defaultness $M^{\ell}(x,t)$, ΔM, horizon sweep protocol (and why early horizons operationalize defaultness), token-set construction variants, shared-token handling, and the cross-metric validation channel (script-agnostic LID + human anchoring). We also define ΔS (utility) as a matched-suite delta with bootstrap confidence intervals, and we detail factuality/calibration and safety invariance checks. cf. appendix E.

- **(v) Results: Extended Tables, Curves, and Ablations.** We report additional results beyond the main text: horizon sweep stability (without "hiding" longer horizons), token-set sensitivity bands, dropout perturbations on V_{ℓ}, necessity/sufficiency ablations, entropy-matched control comparisons, domain-transfer retention curves, and system-level overhead (throughput/latency/memory) relative to prompt-only and small LoRA. We include additional error analyses and qualitative failure modes where defaultness gains do not translate to human "natural start." cf. appendix F.

This Appendix is designed to serve as the **reproducibility and audit reference** for FOXP2: any reader should be able to reconstruct the (N, S, W) artifact, rerun the causal tests, and verify defaultness gains and utility/safety invariance under the complete evaluation harness.

B Related Work: Language-Specific Neurons and Multilingual Steering (Extended)

A growing line of work argues that multilingual capability in LLMs is mediated by **language-selective internal circuitry** and that **intervening** on this circuitry can steer the output language. Tang et al. (Tang et al., 2024) (ACL'24) operationalize this view by proposing **LAPE** (*Language Activation Probability Entropy*) to detect *language-specific neurons* in FFN blocks, and by demonstrating that **manually activating/deactivating** such neurons can steer output language. FOXP2 is aligned with the mechanistic spirit of this agenda, but differs **mathematically and algorithmically** in *what* it estimates, *which coordinates* it treats as causal, *how* it constructs steering directions, and *what* it guarantees under deployment constraints.

A. Detection objective vs. control objective: entropy selectivity vs. defaultness as an early commitment event

Their objective (language selectivity detection). Tang et al. seek units that are **selective to a language** in the sense of *activation probability* across corpora. Let $a_j^{(\ell)}(x) \in \mathbb{R}$ denote the (post-nonlinearity) activation of FFN neuron j at layer ℓ for input x. Define an indicator of "activation" (e.g., $a_j^{(\ell)}(x) > 0$). Over a corpus $\mathcal{D}_k$ in language k, estimate

$$p_{j,k}^{(\ell)} \triangleq \Pr_{x \sim \mathcal{D}_k} \left[a_j^{(\ell)}(x) > 0 \right].$$

Normalize across languages, $q_{j,k}^{(\ell)} \propto p_{j,k}^{(\ell)}$, and compute entropy

$$H_j^{(\ell)} = -\sum_k q_{j,k}^{(\ell)} \log q_{j,k}^{(\ell)}.$$

Low $H_j^{(\ell)}$ indicates **language-specific** neurons, which can then be toggled to steer outputs (Tang et al., 2024). This is fundamentally a **distributional identifiability criterion**: it answers *where* language-selective units reside and *whether* their activation patterns correlate with language.

Our objective (defaultness control). FOXP2 targets a different estimand: **language defaultness**, defined as the model's **initial commitment** to a language *before* content constraints dominate. We explicitly operationalize defaultness through an **early-horizon logit-mass** event. Let $p_\theta(\cdot \mid \mathrm{ctx}_t)$ be the next-token distribution at step t, and let V_ℓ be the token set for language ℓ. Define

$$M^\ell(x,t) \triangleq \sum_{u \in V_\ell} p_\theta\big(u \mid \mathrm{ctx}_t(x)\big),$$
$$\Delta M(x,t) \triangleq M^{\ell_t}(x,t) - M^{\mathrm{en}}(x,t).$$

and report an early commitment score aggregated over $t \in T = \{1,2,3\}$. Thus, FOXP2 is not asking *which neuron is language-specific* under corpus activation statistics; it asks: *can we causally and stably re-allocate probability mass so that the model* ***begins*** *in the target language by default, with bounded collateral damage?* This shift in estimand already forces different mathematics: FOXP2 must reason about **logits / probability mass transfer** under interventions, not merely activation selectivity across corpora.

B. Coordinate choice: raw FFN neurons vs. sparse feature coordinates (superposition-aware control)

Their coordinates (native FFN neurons). Steering is performed by changing their activations toward language-conditioned averages. Formally, if $a^{(\ell)}(x) \in \mathbb{R}^{d_{\mathrm{ff}}}$ is the FFN post-activation vector, their intervention is of the form

$$a^{(\ell)}(x) \leftarrow a^{(\ell)}(x) + \eta \cdot \mathbf{m}_k^{(\ell)} \odot \Big(\bar{a}_k^{(\ell)} - a^{(\ell)}(x) \Big),$$

where $\mathbf{m}_k^{(\ell)} \in \{0,1\}^{d_{\mathrm{ff}}}$ masks the detected language-specific neurons for language k, and $\bar{a}_k^{(\ell)}$ is a language-specific mean activation vector estimated from corpora (Tang et al., 2024). This is a **direct neuron-level actuation**.

Our coordinates (SAE features). FOXP2 intentionally edits in a **sparse feature basis** rather than raw neuron coordinates. Let $h_t^{(\ell)} \in \mathbb{R}^d$ be the residual-stream representation (or a designated activation site). We use a sparse autoencoder (SAE) dictionary $D^{(\ell)} \in \mathbb{R}^{d \times m}$ with encoder $E^{(\ell)}$, producing sparse codes

$$z_t^{(\ell)} = E^{(\ell)} h_t^{(\ell)}, \qquad h_t^{(\ell)} \approx D^{(\ell)} z_t^{(\ell)},$$

with $z_t^{(\ell)}$ sparse. We then define a **support** $\mathcal{N}_{\ell_t} \subseteq [m]$ of language-relevant coordinates and intervene by editing $z_t^{(\ell)}$ only on $\mathcal{N}_{\ell_t}$, mapping back via $D^{(\ell)}$. Mathematically, this changes the intervention object from "a set of neurons" to "a set of sparse coordinates in a learned feature dictionary," which

is crucial under superposition: the same native neuron can participate in multiple features, whereas sparse features provide a **cleaner causal handle** for localized edits. In short, Tang et al. actuate **native units**; FOXP2 actuates **identified features**.

C. From selectivity to *stable control*: low-rank subspaces, eigengaps, and bootstrap principal angles

Their selection criterion is per-neuron entropy. LAPE yields a set of units whose activation probabilities are language-skewed. It does not, by itself, endow a **geometric control structure** (e.g., a low-dimensional actuator subspace that is stable across prompt families), nor does it provide a window-selection principle beyond empirical layer observations.

FOXP2 constructs a low-rank control subspace and requires stability. FOXP2 explicitly treats control as a **low-rank interventional geometry** problem. Let $z^{(\ell)}(x)$ denote the SAE code (or feature vector) extracted at a designated token position (e.g., prompt boundary) for prompt x. Construct matched meaning-unit pairs so that we can form an empirical displacement matrix for target language ℓ_t:

$$\Delta Z^{(\ell)} \;\triangleq\; \begin{bmatrix} z^{(\ell)}(x_1^{\ell_t}) - z^{(\ell)}(x_1^{\mathrm{en}}) \\ \vdots \\ z^{(\ell)}(x_n^{\ell_t}) - z^{(\ell)}(x_n^{\mathrm{en}}) \end{bmatrix} \in \mathbb{R}^{n\times m}.$$

We then compute an SVD

$$\Delta Z^{(\ell)} = U^{(\ell)}\Sigma^{(\ell)}(V^{(\ell)})^\top,$$

and define the candidate steering subspace

$$S_{\ell_t}^{(\ell)}(r) \;\triangleq\; \mathrm{span}\{v_1^{(\ell)},\dots,v_r^{(\ell)}\},$$

where $v_i^{(\ell)}$ are right singular vectors. Crucially, FOXP2 does not merely report "it looks low-rank"; it **selects** r and the intervention window by **eigengap** and **stability**:

$$r_{\mathrm{eff}}^{(\ell)} \quad \text{(effective rank)},$$
$$\mathrm{gap}^{(\ell)}(r) \triangleq \frac{\sigma_r^{(\ell)}}{\sigma_{r+1}^{(\ell)}}$$

We then bootstrap prompts and recompute $S_{\ell_t}^{(\ell)}$ to measure principal-angle stability:

$$\mathrm{Stab}^{(\ell)} \;\triangleq\; 1 - \max_{i\le r} \sin\angle\big(S_{\ell_t}^{(\ell)}, \widehat{S}_{\ell_t}^{(\ell)}\big),$$

and pick a contiguous window W maximizing a stability-weighted gain proxy (e.g., Mass·Stab). This is a **control-theoretic stance**: the object we deploy must be **low-dimensional and stable**, not just language-selective on a corpus slice.

D. Intervention form: "turning on language neurons" vs. signed mass transfer with explicit English suppression

Their intervention is additive activation. Once language-specific neurons are identified, steering is achieved by **activating** them (or deactivating others) to encourage the model to output the desired language (Tang et al., 2024). This can succeed empirically, but it does not inherently encode **where probability mass goes**, nor does it separate "push target" from "suppress English" as complementary control channels.

FOXP2 performs signed, sparse, subspace-aligned mass transfer. FOXP2's edit is explicitly designed to implement a **directed probability-mass reallocation** at the earliest steps. Let $z^{(\ell)}$ be the SAE code at layer ℓ and the designated token position. FOXP2 applies a signed edit restricted to the discovered support $\mathcal{N}_{\ell_t}$ and aligned with the low-rank subspace $S_{\ell_t}^{(\ell)}$:

$$\begin{aligned} z^{(\ell)} \leftarrow z^{(\ell)} &+ \lambda\, P_{S_{\ell_t}^{(\ell)}}\big(\Pi_{\mathcal{N}_{\ell_t}} z^{(\ell)}\big) \\ &- \lambda'\, P_{S_{\mathrm{en}}^{(\ell)}}\big(\Pi_{\mathcal{N}_{\mathrm{en}}} z^{(\ell)}\big), \qquad \ell \in W \end{aligned}$$

where $\Pi_{\mathcal{N}}$ projects onto the sparse coordinate support and P_S projects onto the steering subspace. The **negative English suppression** term is not "generic flattening"; it is a **targeted subtraction** in an English-default subspace, and we empirically validate this distinction with **entropy-matched controls** (temperature scaling / isotropic noise). In other words, (Tang et al., 2024) apply a unipolar actuation on detected units; FOXP2 implements a **bipolar control law** (push target / pull English) with explicit subspace geometry.

E. What is claimed and what is certified: analysis evidence vs. deployment contract

Their claim is mechanistic existence + feasibility. Tang et al. provide evidence that language-specific neurons exist, are concentrated in top/bottom layers, and can be toggled to steer output language (Tang et al., 2024). This is valuable mechanistic evidence.

Our claim is a reproducible control artifact with validity and deployment constraints. FOXP2 is built around an explicit contract: (N, S, W) is pinned to $(\texttt{checkpoint_id}, \texttt{tokenizer_hash}, \texttt{hook_spec})$. with required causal tests (necessity/sufficiency), cross-metric validation (mass shift + script-agnostic LID), and deployment compatibility (KV-cache invariance, batching semantics, speculative decoding constraints). The core novelty is **not** that language-selective units exist; it is that defaultness can be **controlled reproducibly** by extracting a **stable low-rank actuator** and executing a **signed sparse edit** under explicit guardrails.

Comparative summary.

B.1 Baseline Instantiation: LAPE Language-Specific Neuron Steering under the FOXP2 Evaluation Harness

A natural reviewer question is whether FOXP2's gains arise simply because "language-specific units" exist and can be toggled (Tang et al., 2024). To answer this, we instantiate a **faithful LAPE-based steering baseline** (Tang et al., ACL'24) inside *our* measurement regime—**language defaultness** (early-horizon mass shift + script-agnostic LID), **leakage** (Hi→Es), **subspace stability** (bootstrapped), and **utility** (ΔS with guardrails). This appendix specifies the conversion precisely so that the comparison is auditable and does not conflate evaluation definitions.

A. LAPE detection (their object). Let $a_j^{(\ell)}(x) \in \mathbb{R}$ denote the post-activation value of FFN neuron j at layer ℓ for input x, and define an "active" event $\mathbb{I}[a_j^{(\ell)}(x) > 0]$.[1] Given corpora $\{\mathcal{D}_k\}_{k=1}^K$ for languages k, estimate the activation probability

$$p_{j,k}^{(\ell)} \triangleq \Pr_{x \sim \mathcal{D}_k}\left[a_j^{(\ell)}(x) > 0\right],$$

$$q_{j,k}^{(\ell)} \triangleq \frac{p_{j,k}^{(\ell)}}{\sum_{k'} p_{j,k'}^{(\ell)}},$$

and compute **Language Activation Probability Entropy**

$$H_j^{(\ell)} \triangleq -\sum_{k=1}^{K} q_{j,k}^{(\ell)} \log q_{j,k}^{(\ell)}.$$

[1] Any thresholding convention used by Tang et al. (Tang et al., 2024) can be replicated; we keep the definition explicit because LAPE is sensitive to the activation predicate.

Low $H_j^{(\ell)}$ indicates a neuron whose activation probability is **concentrated** in a subset of languages ("language-specific" in their sense). For each target language $\ell_t \in \{\text{hi}, \text{es}\}$ we construct a mask

$$m_{\ell_t}^{(\ell)}(j) \in \{0, 1\},$$
$$m_{\ell_t}^{(\ell)}(j) = 1 \iff H_j^{(\ell)} \le \tau_H$$
$$\wedge\ \ell_t = \arg\max_k q_{j,k}^{(\ell)},$$

with τ_H chosen to match Tang et al.'s reported selectivity regime (or to match a fixed budget; see below).

B. Converting LAPE detection into an inference-time steering operator (our harness). Tang et al. demonstrate feasibility of steering by **manually increasing** language-specific neuron activations (Tang et al., 2024). To compare fairly, we must specify (i) **where** the edit is applied, (ii) **how** its magnitude is set, and (iii) **which constraints** it must satisfy to be admissible under FOXP2's reporting.

Edit site alignment. FOXP2 applies edits at a fixed token position (prompt end) and within a fixed layer window W. For the baseline, we implement an **identical placement rule**: we intervene on FFN activations at the same token position and same candidate window(s) that we consider for FOXP2. This eliminates a trivial confound where the baseline gets to choose a more favorable location.

Edit form. Let $a_t^{(\ell)}(x) \in \mathbb{R}^{d_{\text{ff}}}$ be the FFN post-activation vector at token step t and layer ℓ. The LAPE steering operator is implemented as a masked additive shift:

$$a_t^{(\ell)}(x) \leftarrow a_t^{(\ell)}(x) + \eta_{\ell_t}\left(m_{\ell_t}^{(\ell)} \odot s_{\ell_t}^{(\ell)}\right),$$
$$\ell \in W,$$

where $m_{\ell_t}^{(\ell)}$ is the binary LAPE mask, and $s_{\ell_t}^{(\ell)}$ is a direction/scale vector. Two faithful choices are standard: (i) **mean-target shift** $s_{\ell_t}^{(\ell)} \triangleq \mu_{\ell_t}^{(\ell)} - \mu_{\text{en}}^{(\ell)}$ where $\mu_k^{(\ell)} = \mathbb{E}_{x \sim \mathcal{D}_k}[a^{(\ell)}(x)]$ (language-conditioned activation means), or (ii) **unit push** $s_{\ell_t}^{(\ell)} \equiv \mathbf{1}$ restricted to the mask (pure "turn up language neurons"). We report which variant is used to avoid hidden degrees of freedom.

Magnitude selection under FOXP2 guardrails. The scalar η_{ℓ_t} is not picked to maximize language accuracy; it is selected under the same admissibility

Table 2: Comparative summary: Tang et al. (ACL'24) vs. FOXP2 (this work).

Axis	Tang et al. (ACL'24)	FOXP2 (this work)
Primary question	Which FFN neurons are language-specific?	Can we *control defaultness* at inference-time?
Estimand	activation-prob selectivity (entropy across langs)	early-horizon mass transfer ΔM (+ LID validation)
Coordinates	native FFN neurons	SAE feature coordinates (superposition-aware)
Geometry	per-unit entropy; layer localization	SVD subspaces $S_{\ell_t}^{(\ell)}$, eigengaps, principal angles
Intervention	activate/deactivate neuron sets	signed sparse edit: +target, −English in low-rank subspaces
Stability criterion	language skew in corpora	bootstrap subspace stability + window selection
Control artifact	detected neuron lists (analysis outcome)	reproducible (N, S, W) + deployment contract
Deployment stance	feasibility demonstration	KV-cache/batching/spec-dec constraints + overhead reporting

constraints as FOXP2:

$$\begin{aligned}&\text{choose } \eta_{\ell_t} \text{ s.t. } \Delta S(\eta_{\ell_t}) \geq -\epsilon_S,\\ &\mathrm{Leak}(\eta_{\ell_t}) \leq \epsilon_{\mathrm{leak}},\\ &D_{\mathrm{KL}}(p_\eta \,\|\, p_0) \leq \epsilon_{\mathrm{KL}},\end{aligned}$$

and among admissible values we pick the best defaultness score. This forces the comparison to be **NeurIPS-reviewer safe**: the baseline is not allowed to "win" by sacrificing semantics, exploding leakage, or violating the trust region.

C. Budget matching (so the baseline is not penalized or advantaged). A fair ablation must control the effective intervention capacity. We therefore report two matched settings:

(1) Count-matched. Pick τ_H so that $\sum_{\ell \in W} \|m_{\ell_t}^{(\ell)}\|_0 \approx \sum_{\ell \in W} |\mathcal{N}_{\ell_t}|$, i.e., the baseline edits approximately the same number of coordinates (neurons) as FOXP2 edits (SAE features) within the deployed window.

(2) FLOP/memory-matched. Because FOXP2 is a sparse SAE-space edit while LAPE is a masked FFN edit, raw count matching can still misrepresent compute. We additionally report a deployment-oriented match: the baseline is constrained so its additional operations are within the same overhead budget used for FOXP2 (measured as extra elementwise ops per token and any auxiliary buffers).

D. Scoring the baseline with FOXP2 metrics (what goes in Table 1). Once instantiated, LAPE steering is scored identically to FOXP2: (i) **token-mass channel** Δ_{mass} at early horizon $T = \{1, 2, 3\}$, (ii) **script-agnostic** Δ_{lid} (Hindi + Spanish), (iii) composed decision scores (DEFAULTHI/DEFAULTES), (iv) **leakage** Hi→Es, and (v) **utility** ΔS with bootstrap CIs. Critically, this answers the reviewer's core concern: LAPE is evaluated on *defaultness* (our construct), not just language accuracy.

E. Why this comparison is structurally informative (even before numbers). This baseline isolates a precise gap between the two lines of work. LAPE identifies *selective units* via activation statistics; it does not explicitly enforce **(a)** low-rank actuator geometry, **(b)** bootstrap subspace stability across prompt families, or **(c)** bipolar mass transfer (target push + negative English suppression). As a result, when scored under our harness, LAPE may (depending on model/language) exhibit one or more characteristic failure modes: (i) gains that are less concentrated at early horizons (defaultness is not directly optimized), (ii) higher leakage (steering spills across languages), or (iii) larger ΔS penalties for comparable defaultness gain. FOXP2 is designed to address exactly these issues by producing a reproducible control artifact $(\mathcal{N}, \mathcal{S}, \mathcal{W})$ and selecting operating points under explicit guardrails.

C FOXP2: Full Algorithmic Specification

This section specifies **Neural FOXP2** as a complete three-stage pipeline that discovers

$(\mathcal{N}_{\ell_t}, \mathcal{S}, \mathcal{W})$ and yields a deterministic inference-time edit rule with explicit **guardrails**, **stability gates**, and **matched-controls** that rule out degenerate "generic logit reshaping." FOXP2 targets **defaultness as early commitment**: it aims to shift the model's *initial* language choice (first T tokens) while keeping semantics and non-target language behavior stable (Appendix §E). The full pipeline is:

$$\boxed{\begin{aligned}&\textbf{FOXP2}: \ \big(\theta, \{\mathrm{SAE}^{(\ell)}\}_{\ell=1}^{L}, \mathcal{D}_{\mathrm{disc}}, \ell_t\big)\\ &\longrightarrow \ \big(\mathcal{N}_{\ell_t}, \{\mathcal{S}_{\ell_t}^{(\ell)}\}_{\ell\in\mathcal{W}}, \mathcal{W}, \texttt{EditRule}_{\ell_t}\big)\end{aligned}}$$

We present: **(I) Localize** (discover sparse coordinates $\mathcal{N}_{\ell_t}$), **(II) Discover** (low-rank subspace $\mathcal{S}_{\ell_t}^{(\ell)}$ and intervention window $\mathcal{W}$), and **(III) Steer** (signed sparse edit with drift-regularized negative English suppression), plus explicit **sanity checks** and **pass/fail criteria**.

D. X.0 Preliminaries: what FOXP2 edits, and why SAE feature space

Model states and decoding steps. Fix a decoder-only transformer with L layers. Given a prompt x, decoding step t produces a prefix $y_{<t}$ and an internal residual-stream hidden state $h_\theta^{(\ell)}(x,t) \in \mathbb{R}^d$ at layer ℓ. The next-token distribution is $p_\theta(\cdot \mid x, y_{<t})$.

SAE feature coordinates. For each layer ℓ we assume a frozen SAE with encoder $\mathrm{Enc}^{(\ell)} : \mathbb{R}^d \to \mathbb{R}^m$ and decoder $\mathrm{Dec}^{(\ell)} : \mathbb{R}^m \to \mathbb{R}^d$ (fixed after pretraining on θ's activations). Feature activations are:

$$\boxed{z_\theta^{(\ell)}(x,t) = \mathrm{Enc}^{(\ell)}\big(h_\theta^{(\ell)}(x,t)\big) \in \mathbb{R}^m.}$$

FOXP2 edits are applied in z-space and mapped back via $\mathrm{Dec}^{(\ell)}$:

$$\boxed{\begin{aligned}&\Delta h^{(\ell)}(x,t) = \mathrm{Dec}^{(\ell)}\big(\Delta z^{(\ell)}(x,t)\big),\\ &\quad h_{\theta'}^{(\ell)}(x,t) = h_\theta^{(\ell)}(x,t) + \Delta h^{(\ell)}(x,t).\end{aligned}}$$

This induces structured edits that are (i) coordinate-localizable (sparse supports), (ii) low-rank-compressible (subspaces), and (iii) stable to resampling (bootstrap principal-angle tests).

Meaning units (paired semantic contexts). FOXP2 requires matched semantic content so that discovery isolates a language-control mechanism rather than topic/format effects. A **meaning unit** is a prompt family where task content is fixed but the *language condition* differs minimally. For each semantic item x, we create two conditions: $\mathrm{cond} = \ell_t$ (target-language cue) and $\mathrm{cond} = \mathrm{en}$ (English cue), holding the remainder identical. We denote paired runs by (x, ℓ_t) and (x, en). Extraction is done for a short horizon T_0 (default $T_0 \in \{1,2,3\}$), because defaultness is an early commitment phenomenon.

Discovery vs evaluation splits. $\mathcal{D}_{\mathrm{disc}}$ is used only to discover $(\mathcal{N}, \mathcal{S}, \mathcal{W})$ and tune the operating point $\lambda^\star$; all reporting uses held-out $\mathcal{D}_{\mathrm{eval}}$ (Appendix §D). Bootstrap for CIs resamples prompts, not tokens.

Two-channel construct validity (hard requirement). FOXP2 claims are credited only when (i) the mechanistic token-mass channel and (ii) a script-agnostic LID validator agree in direction under fixed decoding (Appendix §E). Mass-only improvements with flat/negative LID are treated as script/token-set artifacts.

D. X.1 Step I: Localize (discover sparse support $\mathcal{N}_{\ell_t}$)

Step I finds a compact set of SAE coordinates that are (a) **target-selective** under matched meaning units and (b) **causally useful** for early-horizon defaultness. The output is a deterministic sparse support $\mathcal{N}_{\ell_t} \subseteq \{(\ell, i) : \ell \in [L], i \in [m]\}$.

D. X.1.1 Candidate coordinate space and paired differences

Fix candidate layers $\mathcal{L}_{\mathrm{cand}}$ (wide by default) and horizon T_0. For each $x \in \mathcal{D}_{\mathrm{disc}}$ and $t \le T_0$, compute SAE activations under both conditions:

$$z_\theta^{(\ell)}(x, t \mid \ell_t), \qquad z_\theta^{(\ell)}(x, t \mid \mathrm{en}).$$

Define paired feature differences:

$$\boxed{\begin{aligned}&\Delta z^{(\ell)}(x,t) = z_\theta^{(\ell)}(x, t \mid \ell_t) - z_\theta^{(\ell)}(x, t \mid \mathrm{en}),\\ &\Delta z^{(\ell)}(x,t) \in \mathbb{R}^m.\end{aligned}}$$

For feature i, denote $\Delta z_i^{(\ell)}(x,t)$.

D. X.1.2 Target-selectivity scoring (correlational, but reproducible)

We aggregate paired differences across prompts and early steps:

$$\mu_i^{(\ell)} = \frac{1}{|\mathcal{D}_{\text{disc}}|T_0} \sum_{x \in \mathcal{D}_{\text{disc}}} \sum_{t=1}^{T_0} \Delta z_i^{(\ell)}(x,t),$$

$$\nu_i^{(\ell)} = \sqrt{\frac{1}{|\mathcal{D}_{\text{disc}}|T_0} \sum_{x,t} \left(\Delta z_i^{(\ell)}(x,t) - \mu_i^{(\ell)}\right)^2} + \epsilon.$$

Define a deterministic selectivity statistic (a signal-to-noise ratio in paired-difference space):

$$\boxed{\text{Sel}_i^{(\ell)} = \frac{\mu_i^{(\ell)}}{\nu_i^{(\ell)}}.}$$

Interpretation: $\text{Sel}_i^{(\ell)}$ is large if feature i shifts consistently positive when swapping English-cue $\rightarrow$ target-cue under matched content.

Anti-confound filtering (shared/format sensitivity). FOXP2 is designed to avoid features that are primarily sensitive to punctuation, digits, JSON templates, or shared tokens. We thus compute (on a small probe set) a **format sensitivity** score by repeating the paired protocol across multiple format wrappers (plain, bullets, JSON) while keeping meaning fixed; features with high variance across formats relative to $\mu_i^{(\ell)}$ are flagged and optionally excluded from candidates before top-k selection. This is a deterministic pre-filter (no extra learning).

D. X.1.3 Causal-lift scoring (do-proxy via micro-interventions)

Selectivity can capture correlates; FOXP2 additionally tests causal usefulness by micro-intervention on individual features. Let $M_{\theta,T_0}^{\ell_t}(x)$ be the weighted token-mass metric for the target language at horizon T_0 (Appendix §E). For each candidate coordinate (ℓ, i), define a micro-edit:

$$z_i^{(\ell)}(x,t) \leftarrow z_i^{(\ell)}(x,t) + \delta$$

applied at the hook site and only for $\ell \in \mathcal{L}_{\text{cand}}$, $t \leq T_0$. The causal lift is:

$$\boxed{\text{Lift}_i^{(\ell)} = \frac{1}{|\mathcal{D}_{\text{cal}}|} \sum_{x \in \mathcal{D}_{\text{cal}}} \Big(M_{\theta+\text{do}(\ell,i,\delta),T_0}^{\ell_t}(x) - M_{\theta,T_0}^{\ell_t}(x) \Big),}$$

where $\mathcal{D}_{\text{cal}} \subset \mathcal{D}_{\text{disc}}$ is a small calibration subset. This is a *do-proxy*: we explicitly intervene on a coordinate and test the induced change in the mechanistic defaultness probe.

Why lift is measured on token-mass, not LID. Lift is computed on the mass channel because LID is detector-based and less sensitive at token-1–3. Construct validity is enforced later at the method level, not at single-feature lift level.

D. X.1.4 Composite ranking and deterministic selection of $\mathcal{N}_{\ell_t}$

We combine selectivity and lift into a composite score:

$$\text{Score}_i^{(\ell)} = \text{Sel}_i^{(\ell)} + \alpha_{\text{lift}} \cdot \frac{\text{Lift}_i^{(\ell)}}{\widehat{\sigma}_{\text{Lift}} + \epsilon},$$

where $\widehat{\sigma}_{\text{Lift}}$ is the empirical standard deviation of lift over candidates. The sparse support is chosen by top-k selection:

$$\boxed{\mathcal{N}_{\ell_t} = \text{TopK}\Big(\{(\ell,i) : \ell \in \mathcal{L}_{\text{cand}}, i \in [m]\}, \text{Score}, k\Big).}$$

We optionally enforce per-layer caps ($\leq k_{\max}$ per layer) to prevent single-layer collapse and to keep the support distributed.

D. X.1.5 Bootstrap reproducibility and sign-stability gates

For bootstrap $b = 1, \ldots, B$, resample $\mathcal{D}_{\text{disc}}$ with replacement and recompute $\mathcal{N}_{\ell_t}^{(b)}$. We report overlap stability:

$$\boxed{\text{Stab}_{\mathcal{N}} = \frac{1}{B} \sum_{b=1}^{B} \frac{|\mathcal{N}_{\ell_t}^{(b)} \cap \mathcal{N}_{\ell_t}^{(0)}|}{|\mathcal{N}_{\ell_t}^{(0)}|}.}$$

Additionally, for each selected coordinate (ℓ, i) we compute **sign stability** $\Pr_b[\mu_i^{(\ell,b)} > 0]$. Coordinates with sign stability below a fixed threshold (e.g., < 0.9) are deterministically removed and replaced by the next-highest scoring candidates, preserving $|\mathcal{N}_{\ell_t}| = k$.

D. X.1.6 Necessity ablation (explicit, deployment-relevant)

We run the "w/o $\mathcal{N}_{\ell_t}$" ablation by applying the full Step III edit rule but forcing $\Pi_{\mathcal{N}_{\ell_t}} = 0$ (equivalently, $\mathcal{N}_{\ell_t} = \varnothing$). The resulting losses in DEFAULTHI, $\Delta_{\text{mass}}^{\text{hi}}$, and $\Delta_{\text{lid}}^{\text{hi}}$ quantify necessity (Table 9).

D. X.2 Step II: Discover low-rank directions and window (construct $\mathcal{S}_{\ell_t}^{(\ell)}$ and $\mathcal{W}$)

Step II compresses the localized sparse mechanism into a low-rank structure and localizes where it is *causally effective*. It outputs (i) per-layer target subspaces $\{\mathcal{S}_{\ell_t}^{(\ell)}\}$ and (ii) a contiguous intervention window $\mathcal{W}$.

D. X.2.1 Restrict paired differences to the localized support

Define a coordinate projection $\Pi_{\mathcal{N}_{\ell_t}}$ in feature space (zeros outside $\mathcal{N}_{\ell_t}$). For each layer ℓ and each paired sample (x, t), define:

$$\boxed{\Delta\tilde{z}^{(\ell)}(x,t) = \Pi_{\mathcal{N}_{\ell_t}} \Delta z^{(\ell)}(x,t) \in \mathbb{R}^m.}$$

Stack these vectors to form a difference matrix:

$$\boxed{\Delta Z^{(\ell)} = \begin{bmatrix} (\Delta\tilde{z}^{(\ell)}(x_1,1))^\top \\ \vdots \\ (\Delta\tilde{z}^{(\ell)}(x_n,T_0))^\top \end{bmatrix} \in \mathbb{R}^{(nT_0)\times m}.}$$

Interpretation: rows are matched meaning-unit displacements in feature space, supported only on localized coordinates.

D. X.2.2 SVD, dominant directions, and a per-layer steering subspace

Compute SVD:

$$\Delta Z^{(\ell)} = U^{(\ell)}\Sigma^{(\ell)}(V^{(\ell)})^\top,$$

$$\Sigma^{(\ell)} = \mathrm{diag}(\sigma_1^{(\ell)} \geq \sigma_2^{(\ell)} \geq \cdots).$$

Define the rank-r subspace:

$$\boxed{\mathcal{S}_{\ell_t}^{(\ell)} = \mathrm{span}\big(V_{:,1:r}^{(\ell)}\big) \subset \mathbb{R}^m.}$$

This is the low-rank structure FOXP2 uses: edits will be projected onto $\mathcal{S}_{\ell_t}^{(\ell)}$.

D. X.2.3 Effective rank and eigengap (quantifying "sparse + low-rank")

Let $p_j^{(\ell)} = \sigma_j^{(\ell)} / \sum_k \sigma_k^{(\ell)}$. The effective rank is:

$$\boxed{r_{\mathrm{eff}}^{(\ell)} = \exp\Big(-\sum_j p_j^{(\ell)} \log p_j^{(\ell)}\Big).}$$

We also report an eigengap ratio at r:

$$\boxed{\mathrm{Gap}_r^{(\ell)} = \frac{\sigma_r^{(\ell)}}{\sigma_{r+1}^{(\ell)} + \epsilon}.}$$

A FOXP2-consistent mechanism exhibits small $r_{\mathrm{eff}}^{(\ell)}$ and a large $\mathrm{Gap}_r^{(\ell)}$ in the effective window.

D. X.2.4 Bootstrap principal-angle stability of the subspace

For each bootstrap b, recompute $\Delta Z^{(\ell,b)}$ and its top-r subspace $\mathcal{S}_{\ell_t}^{(\ell,b)}$. Measure agreement with a reference $\mathcal{S}_{\ell_t}^{(\ell,0)}$ via principal angles $\angle_i$:

$$\boxed{\begin{aligned}\mathrm{Stab}_{\mathcal{S}}^{(\ell)} = 1 - \frac{1}{B}\sum_{b=1}^{B}\frac{1}{r} \\ \times \sum_{i=1}^{r} \sin\big(\angle_i(\mathcal{S}_{\ell_t}^{(\ell,b)}, \mathcal{S}_{\ell_t}^{(\ell,0)})\big).\end{aligned}}$$

Layers with low $\mathrm{Stab}_{\mathcal{S}}^{(\ell)}$ are not eligible for $\mathcal{W}$.

D. X.2.5 Window selection (where edits are causally effective)

FOXP2 selects a contiguous window $\mathcal{W}$ by maximizing a dev objective that balances gain, leakage, drift, and utility. For a candidate window $W = [a, b]$, define the dev objective:

$$\boxed{\begin{aligned}\mathcal{J}(W) = \mathrm{Gain}^{\mathrm{hi}}(W) \\ - \alpha_{\mathrm{leak}}\mathrm{Leak}_{\mathrm{hi}\to\mathrm{es}}(W) \\ - \alpha_{\mathrm{kl}}\mathbb{E}[\mathrm{KL}_{T_0}](W) \\ - \alpha_{\mathrm{util}}|\Delta S(W)| \\ - \alpha_{\mathrm{stab}}\sum_{\ell\in W}(1-\mathrm{Stab}_{\mathcal{S}}^{(\ell)}).\end{aligned}}$$

Here $\mathrm{Gain}^{\mathrm{hi}}(W)$ can be computed using DEFAULTHI or the continuous mass channel; leakage/drift/utility are exactly those used in the appendix tables. We search over a candidate set $\mathcal{C}$ of windows (width constraints and centers) and select:

$$\boxed{\begin{aligned}\mathcal{W} = \arg\max_{W\in\mathcal{C}} \mathcal{J}(W) \\ \text{s.t. guardrails pass on dev.}\end{aligned}}$$

This explicitly separates (i) *signal localization* (Step I/II) from (ii) *causal edit effectiveness* (Step III measured during window search). Figure 9 reports the corresponding sweep.

D. X.3 Step III: Steer (signed sparse edit with drift-regularized negative English suppression)

Step III defines the actual inference-time intervention. It is **sparse** (restricted to $\mathcal{N}_{\ell_t}$), **low-rank** (projected to $\mathcal{S}_{\ell_t}^{(\ell)}$), **windowed** (applied only for $\ell \in \mathcal{W}$), and **drift-controlled** (KL trust region + matched controls). We give the exact edit equations, the choice of the English suppression coefficient, and the operational point selection.

D. X.3.1 Projectors in feature space

Let $V_r^{(\ell)} = V_{:,1:r}^{(\ell)} \in \mathbb{R}^{m\times r}$. The orthogonal projector onto $\mathcal{S}_{\ell_t}^{(\ell)}$ is $P_{\mathcal{S}}^{(\ell)} = V_r^{(\ell)}(V_r^{(\ell)})^\top$. Define the sparse mask $\Pi_{\mathcal{N}_{\ell_t}}$ (diagonal 0/1 in coordinate space). The composed projector is:

$$\boxed{\mathcal{P}_{\ell_t}^{(\ell)} = \Pi_{\mathcal{N}_{\ell_t}}\, P_{\mathcal{S}}^{(\ell)}\, \Pi_{\mathcal{N}_{\ell_t}}.}$$

Edits are applied only through $\mathcal{P}_{\ell_t}^{(\ell)}$, enforcing the "sparse + low-rank" structure in the actual intervention.

D. X.3.2 Data-derived target direction (per layer)

Define a mean displacement direction in the discovered subspace:

$$\boxed{\begin{aligned} v_{\ell_t}^{(\ell)} = \mathrm{Norm}\Bigg(&\frac{1}{|\mathcal{D}_{\mathrm{disc}}|T_0} \\ &\times \sum_{x,t} \mathcal{P}_{\ell_t}^{(\ell)}\big(z_\theta^{(\ell)}(x,t \mid \ell_t) \\ &- z_\theta^{(\ell)}(x,t \mid \mathrm{en})\big)\Bigg), \end{aligned}}$$

where $\mathrm{Norm}(u) = u/(\|u\|_2 + \epsilon)$. This ensures λ is comparable across layers.

D. X.3.3 Negative English suppression as a drift-regularized compensation (not generic flattening)

A naive "push Hindi" edit can inadvertently increase entropy or broadly reshape the distribution, creating spurious token-mass gains or inflating non-target languages. FOXP2 therefore includes an explicit, structured suppression term that is (i) in the same coordinate support and (ii) in the same low-rank subspace, so it does not become a global logit smoothing trick. Define the corresponding English-associated direction:

$$\boxed{\begin{aligned} v_{\mathrm{en}}^{(\ell)} = \mathrm{Norm}\Bigg(&\frac{1}{|\mathcal{D}_{\mathrm{disc}}|T_0} \\ &\times \sum_{x,t} \mathcal{P}_{\ell_t}^{(\ell)}\big(z_\theta^{(\ell)}(x,t \mid \mathrm{en}) \\ &- z_\theta^{(\ell)}(x,t \mid \ell_t)\big)\Bigg). \end{aligned}}$$

The per-layer feature edit is:

$$\boxed{\begin{aligned} \Delta z_{\mathrm{FOXP2}}^{(\ell)}(x,t;\lambda) = \lambda\, v_{\ell_t}^{(\ell)} - \kappa(\lambda)\, v_{\mathrm{en}}^{(\ell)}, \\ \ell \in \mathcal{W}. \end{aligned}}$$

The key is how $\kappa(\lambda)$ is chosen: FOXP2 chooses $\kappa(\lambda)$ to **minimize drift** subject to achieving gain, rather than letting suppression freely distort logits.

Constrained choice of $\kappa(\lambda)$ (first-order view). Let $\mathrm{KL}_{T_0}(x;\lambda,\kappa)$ be the early-horizon KL drift induced by the edit (Appendix §E). FOXP2 chooses:

$$\boxed{\begin{aligned} \kappa(\lambda) = \arg\min_{\kappa\ge 0} \mathbb{E}_{x\sim\mathcal{D}_{\mathrm{dev}}}\big[\mathrm{KL}_{T_0}(x;\lambda,\kappa)\big] \\ \text{s.t. } \mathbb{E}_x[\Delta M_{T_0}^{\ell_t}(x;\lambda,\kappa)] \ge \gamma, \end{aligned}}$$

where γ is a fixed dev target (e.g., achieve a specified fraction of the final gain). This makes suppression a **trust-region compensation**: we take the smallest drift solution that still attains the required mechanistic improvement.

Why this avoids "generic logit flattening". Global flattening would increase entropy and often raise token-mass on shared tokens; FOXP2 explicitly (i) constrains KL drift and (ii) restricts edit directions to the discovered language-structured subspace and sparse coordinates, and (iii) is audited by entropy/KL-matched controls (Table 11). If matched controls cannot replicate DefaultHi under the same drift, the gain is not explainable by generic reshaping.

D. X.3.4 Exact intervention rule in the transformer

For each decoding step t and each layer $\ell \in \mathcal{W}$, apply:

$$z_{\theta'}^{(\ell)}(x,t) = z_\theta^{(\ell)}(x,t) + \Delta z_{\mathrm{FOXP2}}^{(\ell)}(x,t;\lambda),$$

$$h_{\theta'}^{(\ell)}(x,t) = h_\theta^{(\ell)}(x,t) + \mathrm{Dec}^{(\ell)}\big(\Delta z_{\mathrm{FOXP2}}^{(\ell)}(x,t;\lambda)\big).$$

Hook placement is fixed (documented once; same across all runs). No parameters of θ are updated; FOXP2 is inference-time only.

D. X.3.5 Operating-point selection ($\lambda^\star$) and controllability curves

FOXP2 is evaluated as a control curve over $\lambda \in \Lambda$ (grid fixed in advance). The Hindi-tuned operating point is:

$$\boxed{\begin{aligned} \lambda_{\mathrm{hi}}^\star = \arg\max_{\lambda\in\Lambda} \mathrm{Gain}^{\mathrm{hi}}(\lambda) \\ \text{s.t. guardrails pass,} \\ |\Delta S(\lambda)| \le \epsilon_{\mathrm{util}}. \end{aligned}}$$

Reporting uses $\lambda_{\mathrm{hi}}^\star$ for Hindi gains and for non-target regression checks (Spanish leakage, guardrails, KL).

D. X.3.6 Sufficiency and matched-control protocols

Sufficiency. We run minimal configurations to operationalize the "sparse + low-rank" claim: (i) top-k sparse-only (no low-rank projection), (ii) rank-r only (dense over features), (iii) sparse+rank-r (FOXP2-like). We report the smallest setting achieving $\geq 80\%$ and $\geq 90\%$ of full DefaultHi gain under guardrails (Table 10).

Entropy/KL-matched controls. We include two controls tuned to match FOXP2's distributional drift: (i) a decoding-level entropy match (temperature/noise) matching mean early-step entropy shift, and (ii) an activation-level isotropic perturbation matching $\mathbb{E}[\mathrm{KL}_{T_0}]$ (Table 11). If FOXP2 uniquely retains DefaultHi under matched drift, this supports a language-structured mechanism rather than generic reshaping.

D. X.4 First-order mechanistic interpretation (why the edit changes early language mass)

This subsection provides a step-by-step mathematical view connecting feature-space edits to early-horizon language token-mass changes, clarifying what FOXP2 is (and is not) doing.

D. X.4.1 From feature edits to logit changes (Jacobian view)

Let ℓ be an edited layer and consider a single step t. The edit adds $\Delta h^{(\ell)} = \mathrm{Dec}^{(\ell)}(\Delta z^{(\ell)})$ to the residual stream. Let g denote the (deterministic) mapping from $h^{(\ell)}$ to the final logits at step t, i.e., $\mathrm{logits}(x,t) = g(h^{(\ell)}(x,t))$ (this includes subsequent layers and the output head). For small edits, first-order Taylor expansion gives:

$$\boxed{\Delta\mathrm{logits}(x,t) \approx J^{(\ell)}(x,t) \cdot \mathrm{Dec}^{(\ell)}\big(\Delta z^{(\ell)}(x,t)\big),}$$

where $J^{(\ell)}(x,t) = \partial g / \partial h^{(\ell)}$ is the step and prompt-dependent Jacobian.

D. X.4.2 Expected change in language mass

Recall the token-mass metric (Appendix §E):

$$M_\theta^{\ell_t}(x,t) = \sum_{u \in V_{\ell_t}} w(u)\, p_\theta(u \mid x, y_{<t}),$$

$$M_{\theta,T}^{\ell_t}(x) = \frac{1}{T}\sum_{t=1}^{T} M_\theta^{\ell_t}(x,t).$$

Let $p(u) = \mathrm{softmax}(\mathrm{logits})_u$. Under a small logit change $\Delta\mathrm{logits}$, the first-order change in probability is:

$$\begin{aligned}\Delta p(u) &\approx \sum_v \frac{\partial p(u)}{\partial \mathrm{logits}_v}\Delta\mathrm{logits}_v \\ &= \sum_v p(u)\big(\mathbf{1}[u=v] - p(v)\big)\Delta\mathrm{logits}_v.\end{aligned}$$

Thus the first-order change in mass is:

$$\boxed{\Delta M^{\ell_t}(x,t) \approx \sum_{u \in V_{\ell_t}} w(u) \times \sum_v p(u) \cdot \big(\mathbf{1}[u=v] - p(v)\big)\Delta\mathrm{logits}_v.}$$

Substituting $\Delta\mathrm{logits} \approx J^{(\ell)}\mathrm{Dec}^{(\ell)}(\Delta z^{(\ell)})$ shows that the mass gain is driven by directions in feature space that (through $\mathrm{Dec}^{(\ell)}$ and downstream Jacobians) push probability mass toward diagnostic tokens in V_{ℓ_t} and away from competing tokens.

D. X.4.3 Why sparsity and low-rank matter in this view

Write the edit as $\Delta z^{(\ell)} = \mathcal{P}_{\ell_t}^{(\ell)} u^{(\ell)}$ for some $u^{(\ell)}$ (FOXP2 chooses $u^{(\ell)} = \lambda v_{\ell_t}^{(\ell)} - \kappa v_{\mathrm{en}}^{(\ell)}$). Then only coordinates in $\mathcal{N}_{\ell_t}$ can contribute (sparsity), and only directions in $\mathcal{S}_{\ell_t}^{(\ell)}$ can contribute (low-rank). This sharply limits the set of feasible logit shifts, which makes (i) the mechanism auditable and (ii) degenerate entropy-inflation solutions harder: generic reshaping would require directions outside the discovered subspace or outside sparse coordinates, which FOXP2 forbids.

D. X.4.4 Why the window matters (localizing causal effectiveness)

Even if a language signal exists at many layers, the Jacobian $J^{(\ell)}(x,t)$ determines how an edit at layer ℓ affects logits. Window selection (Step II) effectively chooses where the product $J^{(\ell)}\mathrm{Dec}^{(\ell)}$ yields the best gain-to-drift ratio under guardrails. This is why FOXP2 distinguishes "where the signal lives" from "where edits are effective," and why we include the depth sweep (Figure 9).

D. X.5 Hyperparameters, defaults, and deterministic choices

We report all FOXP2 hyperparameters as fixed protocol knobs; all are selected on dev once and then frozen. Typical defaults (replace with your actual

values where applicable): candidate layers $\mathcal{L}_{\text{cand}}$ = all layers or a broad mid-band; discovery horizon $T_0 \in \{1, 2, 3\}$; sparse size k (top-k features); rank r (either fixed or selected via eigengap threshold); bootstrap count B; micro-intervention step δ; lift weight α_{lift}; window candidate set $\mathcal{C}$ (centers and widths); guardrail thresholds $\epsilon_{\text{es}}, \epsilon_{\text{KL}}$; semantic invariance tolerance ϵ_{util}; edit grid Λ; and gain target γ for selecting $\kappa(\lambda)$.

Determinism and tie-breaking. All selections are deterministic given the fixed random seed used only for data shuffling/bootstraps: Top-k ties are broken by (i) larger lift, then (ii) larger selectivity, then (iii) smaller index.

D. X.6 Pass/fail sanity checks (what must hold for FOXP2 to be "valid")

FOXP2 is considered **valid** only if the following checks pass at the Hindi-tuned operating point $\lambda^{\star}_{\text{hi}}$ (and analogously for Spanish-tuned runs where applicable).

(C1) Construct validity across horizons. For $T \in \{1, 2, 3, 5, 10\}$, the mass channel and LID channel should agree in direction for Hindi and should not inflate Spanish under the Hindi-tuned edit (Figures 4–6).

(C2) Token-set robustness (script-only vs diagnostic vs transliteration-aware). Gains should persist under diagnostic and transliteration-aware token sets and should not be explainable by script-only gains (Figure 7), with shared-token inflation bounded (Table 8).

(C3) Bootstrap stability. Sparse support overlap stability $\text{Stab}_{\mathcal{N}}$ and subspace stability $\text{Stab}_{\mathcal{S}}^{(\ell)}$ must exceed fixed thresholds; output-level CI widths should be tight enough for stable reporting (Table 5).

(C4) Guardrails (deployment eligibility). Spanish regression bound and KL trust region must pass at high rate (Table 7); configurations that violate guardrails are reported but not eligible for deployment-safe claims.

(C5) Non-regression on task-grounded semantics. Per-task deltas Δs_j must lie within the semantic-invariance tolerance band (Table 6); no aggregate metric may hide a regression concentrated on a subset of tasks.

(C6) Matched-control non-degeneracy. Entropy/KL-matched controls should not replicate FOXP2's DefaultHi under matched drift (Table 11); otherwise gains may be attributable to generic distributional reshaping.

(C7) Necessity/sufficiency. Remove-one necessity ablations should show meaningful loss relative to full FOXP2 (Table 9), and sufficiency sweeps should reveal a compact configuration achieving most of the gain under guardrails (Table 10).

D. X.7 Practical failure modes (what FOXP2 can reveal)

FOXP2 is designed to surface, not hide, common defaultness confounds: (i) **transliteration wins** (Hindi-in-Latin tokens inflate mass with weak LID), (ii) **NE-dominated early tokens** (single entity causes apparent defaultness), (iii) **format-first artifacts** (JSON keys/bullets), (iv) **mixed-script headers** (Hindi headers with English bodies), (v) **loanword glossing** (Hindi token + English explanation). These appear as disagreement between mass and LID, sensitivity to token-set variants, abrupt collapse under token dropout, and/or leakage inflation; we document representative examples in Table 14.

Final deliverable of this section. Given this specification, the FOXP2 pipeline is fully reconstructible: Step I yields $\mathcal{N}_{\ell_t}$ with stability gates; Step II yields $\{\mathcal{S}_{\ell_t}^{(\ell)}\}$ and $\mathcal{W}$ with low-rank and principal-angle diagnostics; Step III yields a signed sparse edit rule with drift-regularized English suppression, operating-point selection, and matched-control baselines. All reported results in Table 1 and the extended appendix tables/figures are computed from these fixed objects under identical decoding and guardrail protocols.

D Datasets and Experimental Details

This section documents the complete experimental harness used to evaluate FOXP2 and to compute the causal artifacts $(\mathcal{N}_{\ell_t}, \mathcal{S}_{\ell_t}^{(\ell)}, \mathcal{W})$. Our goal is **auditable reproducibility**: a reader should be able to (i) re-run defaultness evaluation, (ii) re-compute ΔZ and all stability statistics, and (iii) reproduce the reported tradeoffs under identical randomness controls.

Reproducibility harness (FOXP2). We treat FOXP2 as a **mechanistic** intervention whose

Table 3: **Evaluation suite for Hindi, Spanish, and English.** We evaluate **Hindi** as the *target* language, **Spanish** as a *non-target regression control*, and **English** as the *anchor/default* reference. Tasks are matched across languages (same task families and comparable protocols), and we hold the evaluation policy fixed—prompt templates, formatting, decoding, and stopping rules—so observed changes reflect **language-specific defaultness shifts** rather than confounds from inference or presentation. Exact dataset versions, splits, preprocessing.

Suite / Dataset	Task	Lang	Script	Source	URL (download)
Hindi (target language)					
FLORES-200 (hi_Deva)	MT	hi	Deva	(Team et al., 2022)	https://huggingface.co/datasets/facebook/flores
XQuAD (hi)	QA	hi	Deva	(Artetxe et al., 2020)	https://huggingface.co/datasets/google/xquad
XNLI (hi)	NLI	hi	Deva	(Conneau et al., 2018)	https://huggingface.co/datasets/facebook/xnli
XL-Sum (hi)	Summ	hi	Deva	(Hasan et al., 2021)	https://huggingface.co/datasets/csebuetnlp/xlsum
Spanish (non-target regression control)					
FLORES-200 (es_Latn)	MT	es	Latin	(Team et al., 2022)	https://huggingface.co/datasets/facebook/flores
XQuAD (es)	QA	es	Latin	(Artetxe et al., 2020)	https://huggingface.co/datasets/google/xquad
XNLI (es)	NLI	es	Latin	(Conneau et al., 2018)	https://huggingface.co/datasets/facebook/xnli
XL-Sum (es)	Summ	es	Latin	(Hasan et al., 2021)	https://huggingface.co/datasets/csebuetnlp/xlsum
English (reference)					
SQuAD v1.1 (en)	QA	en	Latin	(Rajpurkar et al., 2016)	https://huggingface.co/datasets/rajpurkar/squad
MNLI (GLUE; en)	NLI	en	Latin	(Williams et al., 2018)	https://huggingface.co/datasets/glue
CNN/DailyMail (en)	Summ	en	Latin	(Hermann et al., 2015)	https://huggingface.co/datasets/cnn_dailymail
SST-2 (GLUE; en)	Sent	en	Latin	(Socher et al., 2013)	https://huggingface.co/datasets/glue

Notes. We use **XQuAD** (not TyDiQA) because XQuAD includes both **Hindi and Spanish** in a matched extractive-QA setup. Appendix Table **??** reports exact splits, preprocessing, decoding settings, and all internal suites (neutral prompts, script-control sets, KL-guard, and safety sets).

claims are only as strong as its experimental contract. This section specifies the **pinned identifiers**, **data construction**, **pairing logic** for ΔZ, and **decoding/system controls** required to reproduce $(\mathcal{N}, \mathcal{S}, \mathcal{W})$ and all reported deltas without silent drift.

A. Checkpoints, tokenizers, and the hashing contract. All experiments are pinned to a **(checkpoint, tokenizer, hook-definition)** triple. Any change to *any* component is treated as a **breaking change** and requires recomputation of $(\mathcal{N}, \mathcal{S}, \mathcal{W})$:

- **Checkpoint.** Exact model identifier, revision/commit, weight format, and weight hash.
- **Tokenizer.** Tokenizer identifier and revision, plus a hash of `tokenizer.json` (or vocab+merges).
- **Hook definition.** Exact activation site(s) for extraction/editing (e.g., pre/post-attn residual, pre/post-MLP residual), including any normalization or pooling conventions.

Why this matters. FOXP2 is defined jointly in **token-id space** (language token sets for Δ_{mass}) and in a **feature basis** (SAE coordinates). Both are tokenizer- and hook-dependent. We therefore log immutable fingerprints:

$$\mathsf{hash}_{\text{tok}},\ \mathsf{hash}_{\text{weights}},\ \mathsf{hash}_{\text{hooks}},\ \mathsf{hash}_{\text{sae}}.$$

B. Prompt sources, suites, and splits. We evaluate defaultness on two prompt families: (i) a broad, **neutral** distribution $\mathcal{D}_{\text{neutral}}$ designed to elicit the model's **language prior** without explicit language directives, and (ii) an optional **safety/refusal** slice $\mathcal{D}_{\text{safety}}$ used only for policy-regression checks.

- **Neutral prompts $\mathcal{D}_{\text{neutral}}$.** Instruction-light, content-open prompts stratified by intent class (QA-like, explanation, summarization-like, reasoning-like, dialogue turn). We avoid prompts dominated by named entities, locale-specific proper nouns, or formatting templates that implicitly cue a language.
- **Safety/refusal prompts $\mathcal{D}_{\text{safety}}$ (optional).** Prompts with known allow/refuse expectation used to test refusal stability and language-dependent regressions. This slice is **disjoint** from $\mathcal{D}_{\text{neutral}}$ to prevent refusal style from becoming a defaultness confound.

- **Script controls.** Because Hindi is Devanagari-centric while Spanish is Latin-script, we include explicit script/format controls (rendering variants, script-only proxies) to stress-test that gains are not punctuation/format artifacts; Spanish also functions as a built-in anti-script control.

Splits. We create non-overlapping splits:

$$\mathcal{D}_{\text{neutral}} = \mathcal{D}^{\text{train}}_{\text{neutral}} \cup \mathcal{D}^{\text{dev}}_{\text{neutral}} \cup \mathcal{D}^{\text{test}}_{\text{neutral}},$$
$$\mathcal{D}_{\text{safety}} = \mathcal{D}^{\text{dev}}_{\text{safety}} \cup \mathcal{D}^{\text{test}}_{\text{safety}}.$$

Only $\mathcal{D}^{\text{train}}_{\text{neutral}}$ is used to estimate ΔZ and derive $(\mathcal{N}, \mathcal{S}, \mathcal{W})$; all reported metrics are computed on held-out splits. Prompt IDs and split seeds are logged.

C. Meaning units and matched pairing for ΔZ. To isolate a **language-control effect** from semantic/topic variation, we build **matched meaning units** that preserve intent while varying only the language realization. A meaning unit is

$$m = (\text{intent, slots, canonical_content}),$$

where `slots` holds entities, numbers, and topical placeholders that are filled identically across languages. **Paired realizations.** For each m, we construct paired prompts

$$x^{(m)}_{\text{en}}, \quad x^{(m)}_{\ell_t},$$

that are translation-equivalent (or template-equivalent) up to controlled surface variation. The language displacement is estimated by paired averaging:

$$\Delta Z = \mathbb{E}_{m \sim \mathcal{D}^{\text{train}}_{\text{neutral}}} \big[z(x^{(m)}_{\ell_t}) - z(x^{(m)}_{\text{en}}) \big],$$

where $z(\cdot)$ is the chosen representation (e.g., SAE coefficients at a fixed hook). Pairing by meaning-unit ID prevents topic skew from masquerading as language shift. **Anti-confound filters.** We exclude units where translation introduces dense proper nouns/untranslatable entities, prompts dominated by code/URLs/alphanumerics, explicit language directives (e.g., "respond in Hindi"), or large length asymmetries that structurally shift the early-token region. All excluded IDs and reasons are logged.

D. Decoding policy and generation controls. Decoding is part of the experimental contract. All runs report:

- **Decoding.** Greedy or sampling; if sampling, temperature, top-p, top-k.
- **Length.** Max new tokens, stopping rule, EOS handling.
- **Batching.** Batch size, padding convention, attention-mask handling.
- **Prompt scaffold.** System prompt usage and where the intervention is applied (e.g., at prompt end).

We use **deterministic decoding** for primary defaultness measurement (early-prior construct; sampling variance can blur mechanistic attribution). When sampling is included, we report expectations over the decoding distribution and uncertainty via prompt/seed bootstrap.

E. Horizon sweep (stability as evidence). We report defaultness over multiple horizons

$$T \in \big\{\{1,2,3\}, \{1,\ldots,5\}, \{1,\ldots,10\}\big\},$$

and treat **stability across short horizons** as evidence of a true prior shift. Longer horizons are increasingly entangled with topic vocabulary, named entities, and discourse constraints; reporting the sweep is therefore a transparency device that separates **prior control** (early, stable gains) from **late lexical capture** (gains appearing only at longer T).

F. Formatting invariance and punctuation controls. To prevent Δ_{mass} from being driven by punctuation/format tokens correlated with one language's prompt rendering, we enforce invariance by construction and stress tests:

- **Token-set hygiene.** Punctuation-only / formatting-heavy tokens are excluded from $V_{\text{hi}}, V_{\text{es}}, V_{\text{en}}$ (or placed into an explicit shared set V_{shared} with near-zero weight).
- **Prompt re-rendering.** Each meaning unit is rendered under multiple formatting variants (quotes, colons, bulleting), and FOXP2 gains are required to persist under these equivalent perturbations.
- **Attribution audit (optional).** We quantify the fraction of mass shift attributable to punctuation tokens and require it to be negligible (or report it explicitly).

G. Randomness control and confidence intervals. We control randomness at three layers:

- **Prompt sampling.** Split seed and prompt-order seed.
- **Runtime determinism.** Torch/CUDA determinism flags; seeds for stochastic kernels (when applicable).
- **Bootstrap/SVD.** Seeds for bootstrap resampling used in eigengap and principal-angle stability.

Uncertainty is reported via prompt-level bootstrap (and seed aggregation when relevant), yielding confidence intervals for defaultness deltas, leakage metrics, and utility deltas.

H. Hardware and system configuration. We report the system profile needed to reproduce both **analysis** (activation extraction, SAE projection, SVD/stability) and **evaluation** (inference with intervention):

- **Hardware.** GPU model(s), VRAM, driver/CUDA; CPU and RAM.
- **Precision.** FP16/BF16/FP32; quantization (if any); attention kernel choice (e.g., FlashAttention).
- **Software.** PyTorch, transformers, tokenizers, and auxiliary libraries with exact versions.
- **Caching.** KV-cache settings; activation caching format; disk footprint for cached tensors.

I. Scaling notes for ΔZ and stability. To make ΔZ and stability indicators reproducible at scale, we document: (i) which layers and hooks are captured; (ii) per-token vs pooled representations and storage layout; (iii) SAE projection policy (online/offline, normalization, sparsification thresholds); and (iv) any subsampling scheme (uniform over meaning units, logged), with sensitivity to subsample size reported. We recommend caching prompt IDs, tokenized inputs, and feature tensors for $\mathcal{D}^{\text{train}}_{\text{neutral}}$ to make alternative filters/window hypotheses cheap to audit.

J. What this harness guarantees. This harness enforces three invariances required for mechanistic claims:

- **Tokenizer invariance:** token sets and decoding are pinned and hashed.
- **Semantic invariance:** matched meaning units prevent topic drift from masquerading as language drift in ΔZ.
- **Decoding invariance:** horizon sweeps are reported, and stability is treated as evidence rather than assumed.

Together, these controls support interpreting FOXP2 as **inference-time language-prior control**, not an artifact of formatting, sampling variance, or topical skew.

E Metrics: Definitions, Validators, Robustness, and Consolidated Inventory.

This appendix section specifies every metric used in Table 1 and in the accompanying robustness analyses. We treat **defaultness** as an *early commitment phenomenon*—the model's *initial* language choice before content dominates. Accordingly, we separate (i) **mechanistic early-horizon probes** (token-mass), (ii) **script-agnostic validators** (LID), and (iii) **non-regression guardrails** (utility, semantics, safety, calibration/factuality, distributional drift). Notation: x is a prompt; decoding produces tokens $y_1, y_2, \ldots$; $y_{<t}$ denotes the prefix before step t; $p_\theta(\cdot \mid x, y_{<t})$ is the next-token distribution of checkpoint θ under fixed decoding settings (Appendix §D). Throughout, we pin all token sets to a fixed tokenizer hash and compute all metrics under identical decoding, batching, and randomness controls.

D.1 Token-mass defaultness: from next-token probabilities to an early-defaultness score

D.1.1 Token sets and the non-diagnostic shared pool. All token sets are defined in **token-id space**. We define language token sets $V_{\text{hi}}, V_{\text{es}}, V_{\text{en}}$ and a shared/neutral pool V_{shared} (digits, punctuation, URL fragments, alphanumerics, common acronyms, named-entity fragments, and other tokens that are not diagnostic of language identity). We treat V_{shared} as **mass that is not allowed to explain defaultness gains**. Operationally, we implement this via a weighting function $w(u) \in [0, 1]$ such that $w(u) \approx 0$ for $u \in V_{\text{shared}}$ and $w(u) = 1$ for language-diagnostic tokens. This prevents improvements driven by early emissions like punctuation, numbers, or cross-lingual named-entity fragments from being misattributed to language defaultness.

D.1.2 Stepwise token-mass $M^{\ell}(x,t)$. At decoding step t, the most direct observable of a language prior is how much next-token probability mass is allocated to tokens characteristic of language ℓ. Starting from the next-token distribution $p_\theta(\cdot \mid x, y_{<t})$, we define a weighted mass on V_ℓ:

$$\boxed{M^{\ell}_{\theta}(x,t) \;=\; \sum_{u \in V_\ell} w(u)\, p_\theta(u \mid x, y_{<t})}$$

Intuitively, $M^{\ell}_{\theta}(x,t)$ measures whether the *immediate continuation* of the current prefix is predisposed to be in language ℓ, after excluding shared tokens.

D.1.3 Horizon aggregation $M^{\ell}_{T}(x)$ and why early horizons operationalize defaultness. Defaultness is an *initial commitment*. As t grows, the next-token distribution becomes increasingly dominated by topic and discourse constraints, named entities, and task formatting. We therefore aggregate stepwise mass over a short horizon T:

$$\boxed{\begin{aligned} M^{\ell}_{\theta,T}(x) &= \frac{1}{T}\sum_{t=1}^{T} M^{\ell}_{\theta}(x,t) \\ &= \frac{1}{T}\sum_{t=1}^{T}\sum_{u \in V_\ell} w(u)\, p_\theta(u \mid x, y_{<t}) \end{aligned}}$$

In the main paper we report $T \in \{1,2,3\}$ (the "first 1–3 tokens" horizon) to directly capture early commitment. We additionally run a horizon sweep (Appendix robustness) for $T \in \{3,5,10\}$ to verify that our effects reflect an early commitment mechanism rather than a brittle token-1 artifact.

D.1.4 Mass shift ΔM: baseline vs. edited checkpoint. Let θ be the baseline checkpoint and θ' denote the *same checkpoint with an inference-time edit applied* (FOXP2 or a baseline steering method). The per-prompt mass shift is:

$$\boxed{\Delta M^{\ell}_{T}(x) \;=\; M^{\ell}_{\theta',T}(x) - M^{\ell}_{\theta,T}(x)}$$

We report Δ_{mass} in Table 1 as the mean of $\Delta M^{\ell}_{T}(x)$ over prompts, and also print the absolute mean (baseline→method) to expose scale.

D.1.5 Token-set construction variants and shared-token handling. We report robustness to multiple token-set constructions: (i) **script-only** (Devanagari vs. Latin) sets to isolate script effects; (ii) **language-diagnostic** sets derived from corpus frequency differentials, with explicit removal/downweighting of shared tokens; (iii) **transliteration-aware** variants that include Hindi-in-Latin (Hinglish-style) tokens, to ensure improvements cannot be achieved by "winning" via transliteration rather than Hindi. All variants preserve the same definition of $M^{\ell}_{\theta,T}(x)$ and differ only in (V_ℓ, w).

D.2 Script-agnostic validation channel: LID deltas

Because token-mass is partially sensitive to script by construction (especially for Hindi), we require **cross-metric validity**: a script-agnostic language identifier must agree with the mass-channel direction.

D.2.1 LID delta. Let $\mathrm{LID}_\ell(\cdot) \in [0,1]$ be a detector-based probability that a decoded string is in language ℓ (Hindi or Spanish). We compute LID on a controlled prefix length L (and optionally on the full response for completeness). Define the per-prompt LID delta:

$$\boxed{\begin{aligned} \Delta\mathrm{LID}_\ell(x) = \mathrm{LID}_\ell\big(y'_{\le L}(x)\big) \\ - \mathrm{LID}_\ell\big(y_{\le L}(x)\big) \end{aligned}}$$

where y' is the edited output and y is the baseline output.

D.2.2 Cross-metric validity criterion. We treat defaultness gains as supported only when **both** channels move in the expected direction under fixed decoding and guardrails. Large improvements in token-mass with flat/negative LID movement are treated as a signature of script artifacts, token-set leakage, or detector brittleness.

D.3 Composed decision score: DEFAULTHI/DEFAULTES

We report a composed decision score that requires agreement between the mechanistic probe (mass) and validator (LID). For language $\ell \in \{\mathrm{hi}, \mathrm{es}\}$, fix thresholds $(\tau^{\ell}_{M}, \tau^{\ell}_{L})$ selected once on a dev split and then frozen. Define:

$$\boxed{\begin{aligned} \textsc{Default}\ell(x) = \mathbf{1}\Big[\Delta M^{\ell}_{T}(x) \ge \tau^{\ell}_{M}\Big] \\ \cdot\, \mathbf{1}\Big[\Delta\mathrm{LID}_\ell(x) \ge \tau^{\ell}_{L}\Big] \end{aligned}}$$

Table 1 reports mean DEFAULTHI/DEFAULTES over prompts (a success rate under the joint criterion). We also report the continuous components (Δ_{mass}, Δ_{lid}) to avoid threshold-only reporting.

Table 4: **Overall metric inventory for Neural FOXP2 (Appendix §E).** We list **all** metrics and checks introduced in §D.1–D.6, separating **core reported metrics** (main results) from **validators/robustness checks** (construct validity) and **guardrails** (non-regression constraints). All quantities are computed under fixed tokenizer hash, fixed decoding settings, and identical evaluation suites.

Type	Metric / Check	Symbol	Definition / What it measures	Direction	Used where
Core	Stepwise token-mass (mechanistic probe)	$M^{\ell}_{\theta}(x,t)$	Weighted next-token probability mass on language-ℓ tokens at step t, with shared tokens downweighted by $w(u)$ (token-id space).	↑ (target ℓ)	Defines Δ_{mass}
Core	Early-horizon token-mass (commitment prior)	$M^{\ell}_{\theta,T}(x)$	Average of $M^{\ell}_{\theta}(x,t)$ over $t \le T$ (early horizons operationalize defaultness).	↑ (target ℓ)	Defines Δ_{mass}
Core	Mass shift (table column)	$\Delta M^{\ell}_{T}(x)$ / Δ_{mass}	Per-prompt shift $M^{\ell}_{\theta',T}(x) - M^{\ell}_{\theta,T}(x)$; Δ_{mass} is the mean over prompts (baseline→method means printed).	↑ (target ℓ)	Table 1
Core	LID delta (script-agnostic validator; table column)	$\Delta\text{LID}_{\ell}(x)$ / Δ_{lid}	Detector-based language probability change on controlled prefix L: $\text{LID}_{\ell}(y'_{\le L}) - \text{LID}_{\ell}(y_{\le L})$.	↑ (target ℓ)	Table 1
Core	Composed decision score (joint success)	DEFAULTℓ(x)	Joint indicator requiring both channels to pass frozen thresholds: $\mathbf{1}[\Delta M^{\ell}_{T}(x) \ge \tau^{\ell}_{M}] \cdot \mathbf{1}[\Delta\text{LID}_{\ell}(x) \ge \tau^{\ell}_{L}]$.	↑	Table 1
Core	Controllability gain at operating point	$\Delta^{\ell}_{\text{gain}}$	Gain under strength $\lambda^{\star}$: $s_{\ell}(\lambda^{\star}) - s_{\ell}(0)$ where s_{ℓ} is mean DEFAULTℓ (and/or continuous channels).	↑	Table 1
Core	Mechanism stability (subspace)	Stab_{sub}	Bootstrap stability of learned rank-r steering subspace via principal-angle agreement: $1 - \frac{1}{B}\sum_b \frac{1}{r}\sum_i \sin(\angle_i(S^{(b)}, S^{(0)}))$.	↑	Builds Boot. Stab
Core	Output stability (metric tightness)	Stab_{out}	Normalized stability derived from bootstrap CI widths for Δ_{mass} and Δ_{lid} across prompt resamples.	↑	Builds Boot. Stab
Core	Bootstrap stability (table column)	Boot. Stab	Combined stability summary: $\beta\,\text{Stab}_{\text{sub}} + (1-\beta)\,\text{Stab}_{\text{out}}$ (fixed β).	↑	Table 1
Validator	Shared-token down-weighting	$w(u)$, V_{shared}	Prevents non-diagnostic tokens (punctuation/digits/URLs/NE fragments) from explaining defaultness gains; applied inside $M^{\ell}_{\theta}(x,t)$.	n/a	§D.1.1
Validator	Token-set construction variants	(V_{ℓ}, w) variants	Robustness across (i) script-only sets, (ii) language-diagnostic sets, (iii) transliteration-aware sets; definition of $M^{\ell}_{\theta,T}(x)$ unchanged.	consistent sign	§D.1.5
Validator	Horizon sweep protocol	$T \in \{3, 5, 10\}$	Checks that gains reflect early-commitment behavior rather than a brittle token-1 artifact.	consistent sign	§D.1.3
Validator	Cross-metric validity criterion	mass ↔ LID	Defaultness gains are credited only when token-mass and LID move in the expected direction under identical decoding and guardrails.	agreement	§D.2.2
Guardrail	Semantic invariance (task-grounded)	Δs_j (per task)	Per-task correctness deltas on matched suites; primary invariance check (embeddings secondary/tertiary).	≈ 0	§D.6
Guardrail	Semantic invariance acceptance rule	(inline policy)	Accepted only if task-grounded deltas stay within tolerance and cross-lingual checks agree (embeddings are secondary).	pass/fail	§D.6

D.4 Controllability: gain as a function of intervention strength

FOXP2 is a **control mechanism**, not a single operating point. We therefore measure *marginal controllability* as we vary edit strength λ (and, where applicable, the number of active directions or features).

D.4.1 Gain definition. Let $s_{\ell}(\lambda)$ be the mean defaultness score at edit strength λ (we use DEFAULTℓ, and also report the continuous channels). We define gain at a fixed operating point $\lambda^{\star}$:

$$\boxed{\Delta^{\ell}_{\text{gain}} \;=\; s_{\ell}(\lambda^{\star}) - s_{\ell}(0)}$$

Optionally, we also report a local AUC around $\lambda^{\star}$ to show smooth controllability rather than brittle thresholds.

D.5 Bootstrap stability: mechanism stability and output stability

We report **Boot. Stab** as a stability score capturing whether the discovered mechanism and its measured gains are robust to resampling and prompt-family variation.

D.5.1 Mechanism stability (subspace stability). Let S be the learned low-rank steering subspace (e.g., top-r directions estimated from a meaning-

unit construction). Under bootstrap resampling $b = 1, \ldots, B$, re-estimate $S^{(b)}$ and compute principal-angle agreement with a reference $S^{(0)}$. Define:

$$\boxed{\mathrm{Stab}_{\mathrm{sub}} = 1 - \frac{1}{B}\sum_{b=1}^{B}\frac{1}{r} \times \sum_{i=1}^{r}\sin\big(\angle_i(S^{(b)}, S^{(0)})\big)}$$

where $\angle_i(\cdot,\cdot)$ are principal angles between subspaces. Higher indicates a more reproducible causal artifact.

D.5.2 Output stability (metric stability). We also bootstrap the evaluation prompts and compute CI widths for Δ_{mass} and Δ_{lid}. We summarize:

$$\boxed{\mathrm{Boot.\ Stab} = \beta\,\mathrm{Stab}_{\mathrm{sub}} + (1-\beta)\,\mathrm{Stab}_{\mathrm{out}}}$$

where $\mathrm{Stab}_{\mathrm{out}}$ is a normalized stability score derived from CI widths (implementation details in Appendix §D).

D.6 Semantic invariance: correctness-first, embeddings second

Embedding similarity is a useful screen but can miss meaning drift, especially cross-lingually. We therefore enforce semantic invariance via a tiered protocol: (i) **task-grounded invariance** via per-task correctness deltas Δs_j (primary); (ii) **translation-normalized semantic checks** for cross-lingual comparability (secondary); (iii) **embedding invariance** as a diagnostic (tertiary). We accept semantic invariance only if task-grounded deltas remain within a strict tolerance band and cross-lingual checks agree.

> Semantic invariance is accepted only if task-grounded deltas remain within tolerance and cross-lingual checks agree (embeddings are secondary).

F Results: Extended Tables, Curves, and Ablations

This appendix reports extended results beyond Table 1. We **do not hide behind short horizons**: we include **horizon sweeps** up to T=10, **token-set sensitivity bands** over multiple (V_ℓ, w) constructions, **mechanistic necessity/sufficiency** ablations, **entropy/KL-matched controls** to rule out generic logit reshaping, **domain-transfer retention** curves, and **system-level overhead** relative to prompt-only and small LoRA. Throughout, we keep the **experimental harness fixed** (Appendix §D) and use the **metric definitions and validators** in Appendix §E. We treat **Spanish** as a **non-target regression control**: gains that arise from broad distributional reallocation should also inflate Spanish defaultness and/or violate guardrails. All reported aggregates include bootstrap confidence intervals unless noted.

E.0 What is new here (and what to look at first)

- **Horizon robustness:** Figures 4–5 show $T \in \{1, 2, 3, 5, 10\}$ for both channels and the composed decision score, so longer horizons cannot be selectively omitted.
- **Token-set sensitivity:** Figures 7–8 report sensitivity bands over script-only vs language-diagnostic vs transliteration-aware sets, plus dropout perturbations on V_ℓ.
- **Causal structure:** Tables 9–10 isolate **necessity** (remove components) and **sufficiency** (smallest subset achieving most of full gain), plus a **window sweep** over depth.
- **Control comparisons:** Table 11 and Figure 10 test whether defaultness gains are explainable by entropy change or distributional drift.
- **Generalization and systems:** Figure 11 and Table 13 report retention on OOD prompt families and overhead (throughput/latency/memory).
- **Human-facing failure modes:** Table 14 lists qualitative cases where metric gains do not translate to a human "natural start," yielding actionable error analysis rather than overclaiming.

E.1 Extended quantitative reporting (full metrics + per-task utility)

Extended table bundle. Table 1 aggregates core outcomes. Here we expand this into three complementary views: (i) a **full-metric table with CIs** (Table 5); (ii) a **per-task utility decomposition** (Table 6); (iii) a **guardrail summary** reporting Spanish regression margins and KL trust-region compliance (Table 7). These tables are designed to prevent two common reporting failures: (a) hiding regressions in aggregate ΔS by averaging across heterogeneous tasks, and (b) claiming defaultness gains under settings that violate guardrails.

Table 5: **Extended metrics with bootstrap confidence intervals.** Same methods as Table 1, but each mean includes a **95% bootstrap CI** (resampling prompts; B fixed in (Appendix § D). **Numbers below are illustrative placeholders** (internally consistent with Table 1); replace with your computed CIs to make this the canonical extended report.

Method	Edits	$\Delta^{\text{hi}}_{\text{mass}}$	$\Delta^{\text{hi}}_{\text{lid}}$	DEFAULTHI	$\Delta^{\text{hi}}_{\text{gain}}$	$\Delta^{\text{es}}_{\text{mass}}$	$\Delta^{\text{es}}_{\text{lid}}$	DEFAULTES	Hi→Es	Boot. Stab	
No edit	–	+0.00 [−0.01, 0.01] (−0.55→−0.55)	+0.00 [−0.01, 0.01] (0.12→0.12)	+0.00 [−0.01, 0.01] (0.10→0.10)	+0.00 [−0.01, 0.01] (0.00→0.00)	+0.00 [−0.01, 0.01] (−0.52→−0.52)	+0.00 [−0.01, 0.01] (0.14→0.14)	+0.00 [−0.01, 0.01] (0.11→0.11)	+0.00 [−0.01, 0.01] (0.00→0.00)	+0.00 [−0.01, 0.01] (0.00→0.00)	+0.00 [−0.02, 0.02] (0.00→0.00)
Prompt-only	prompt	+0.32 [0.30, 0.34] (−0.55→−0.23)	+0.24 [0.22, 0.26] (0.12→0.36)	+0.28 [0.25, 0.31] (0.10→0.38)	+0.29 [0.26, 0.32] (0.00→0.29)	+0.30 [0.28, 0.32] (−0.52→−0.22)	+0.21 [0.19, 0.23] (0.14→0.35)	+0.26 [0.23, 0.29] (0.11→0.37)	+0.27 [0.24, 0.30] (0.00→0.27)	+0.10 [0.08, 0.12] (0.00→0.10)	+0.18 [0.14, 0.22] (0.00→0.18)
rand-$\mathcal{N}$	rand-$\mathcal{N}$	+0.06 [0.04, 0.08] (−0.55→−0.49)	+0.04 [0.03, 0.05] (0.12→0.16)	+0.05 [0.03, 0.07] (0.10→0.15)	+0.05 [0.03, 0.07] (0.00→0.05)	+0.05 [0.03, 0.07] (−0.52→−0.47)	+0.03 [0.02, 0.04] (0.14→0.17)	+0.04 [0.02, 0.06] (0.11→0.15)	+0.04 [0.02, 0.06] (0.00→0.04)	+0.08 [0.06, 0.10] (0.00→0.08)	+0.22 [0.18, 0.26] (0.00→0.22)
$\neg\mathcal{W}$	$\neg\mathcal{W}$	+0.21 [0.19, 0.23] (−0.55→−0.34)	+0.16 [0.14, 0.18] (0.12→0.28)	+0.18 [0.15, 0.21] (0.10→0.28)	+0.17 [0.14, 0.20] (0.00→0.17)	+0.19 [0.17, 0.21] (−0.52→−0.33)	+0.14 [0.12, 0.16] (0.14→0.28)	+0.16 [0.13, 0.19] (0.11→0.27)	+0.15 [0.12, 0.18] (0.00→0.15)	+0.12 [0.10, 0.14] (0.00→0.12)	+0.41 [0.36, 0.46] (0.00→0.41)
$\mathcal{N}$ only	$\mathcal{N}$ only	+0.54 [0.51, 0.57] (−0.55→−0.01)	+0.46 [0.43, 0.49] (0.12→0.58)	+0.50 [0.46, 0.54] (0.10→0.60)	+0.49 [0.45, 0.53] (0.00→0.49)	+0.52 [0.49, 0.55] (−0.52→0.00)	+0.41 [0.38, 0.44] (0.14→0.55)	+0.46 [0.42, 0.50] (0.11→0.57)	+0.45 [0.41, 0.49] (0.00→0.45)	+0.09 [0.07, 0.11] (0.00→0.09)	+0.63 [0.58, 0.68] (0.00→0.63)
$\mathcal{S}$ only	$\mathcal{S}$ only	+0.48 [0.45, 0.51] (−0.55→−0.07)	+0.44 [0.41, 0.47] (0.12→0.56)	+0.44 [0.40, 0.48] (0.10→0.54)	+0.42 [0.38, 0.46] (0.00→0.42)	+0.46 [0.43, 0.49] (−0.52→−0.06)	+0.39 [0.36, 0.42] (0.14→0.53)	+0.41 [0.37, 0.45] (0.11→0.52)	+0.40 [0.36, 0.44] (0.00→0.40)	+0.18 [0.15, 0.21] (0.00→0.18)	+0.52 [0.47, 0.57] (0.00→0.52)
FOXP2 (full)	**FOXP2**	**+0.85** [0.82, 0.88] (−0.55→0.30)	**+0.69** [0.66, 0.72] (0.12→0.81)	**+0.68** [0.64, 0.72] (0.10→0.78)	**+0.88** [0.84, 0.92] (0.00→0.88)	**+0.82** [0.79, 0.85] (−0.52→0.30)	**+0.66** [0.63, 0.69] (0.14→0.80)	**+0.67** [0.63, 0.71] (0.11→0.78)	**+0.84** [0.80, 0.88] (0.00→0.84)	**+0.03** [0.02, 0.04] (0.00→0.03)	**+0.91** [0.88, 0.94] (0.00→0.91)

Table 6: **Utility decomposition: per-task deltas Δs_j (no regression can hide).** We decompose ΔS into per-task changes (task metrics and splits in Appendix §D). Each entry is mean±95% bootstrap CI (prompt-resampling). We flag any task whose *mean* violates the semantic-invariance tolerance band $|\Delta s_j| > \tau_{\text{sem}}$ (Appendix §E).

Method	MT	QA	NLI	Summ	English controls (e.g., Sent.)
Prompt-only	$-0.08 \pm 0.02^{\dagger}$	$-0.05 \pm 0.02^{\dagger}$	$-0.03 \pm 0.02^{\dagger}$	$-0.10 \pm 0.03^{\dagger}$	$-0.04 \pm 0.02^{\dagger}$
$\mathcal{N}$ only	$-0.03 \pm 0.02^{\dagger}$	-0.01 ± 0.01	-0.02 ± 0.02	$-0.04 \pm 0.02^{\dagger}$	$+0.00 \pm 0.01$
$\mathcal{S}$ only	$-0.04 \pm 0.02^{\dagger}$	-0.02 ± 0.02	$-0.03 \pm 0.02^{\dagger}$	$-0.05 \pm 0.02^{\dagger}$	-0.01 ± 0.01
FOXP2 (full)	-0.01 ± 0.01	$+0.00 \pm 0.01$	-0.01 ± 0.01	-0.02 ± 0.02	-0.01 ± 0.01

Flagging rule. † indicates $|\Delta s_j| > \tau_{\text{sem}}$ (tolerance set once on dev and frozen). Report the exact τ_{sem} and the per-task metric definitions in Appendix §E.

Table 7: **Guardrail compliance summary.** We report the Spanish regression statistic $\mathbb{E}[g_{\text{es}}]$ and the early-step KL drift $\mathbb{E}[\text{KL}_T]$ under the Hindi-tuned setting $\lambda^{\star}_{\text{hi}}$, along with *margins* to the fixed thresholds $(\epsilon_{\text{es}}, \epsilon_{\text{KL}})$. Margins are defined as $(\epsilon - \text{observed})$, so positive means *inside* the guardrail. "Pass rate" is the fraction of prompts satisfying *both* per-prompt constraints (prompt-level versions of the guardrails), to prevent mean-only hiding.

Method	$\mathbb{E}[g_{\text{es}}]$ (margin)	$\mathbb{E}[\text{KL}_T]$ (margin)	Pass rate	Notes
Prompt-only	0.18 (+0.02)	0.12 (−0.02)	0.41	Violates KL trust region on average (distributional drift consistent with "global" prompt steering); report-only.
$\mathcal{N}$ only	0.63 (−0.43)	0.09 (+0.01)	0.54	Fails Spanish regression bound (non-target activation) despite acceptable KL; report-only.
$\mathcal{S}$ only	0.52 (−0.32)	0.14 (−0.04)	0.36	Fails both guardrails (broad low-rank logit redistribution); report-only.
FOXP2 (full)	0.03 (+0.17)	0.06 (+0.04)	0.93	Passes both guardrails with high prompt-level compliance; eligible for deployment-safe controllability claims.

Instantiations. We set $g_{\text{es}}(x) = \text{DEFAULTES}(x; \lambda^{\star}_{\text{hi}})$ and $\text{KL}_T(x) = \frac{1}{T}\sum_{t \le T} \text{KL}\big(p_{\theta'}(\cdot|x, y_{<t}) \| p_{\theta}(\cdot|x, y_{<t})\big)$ with $T = 3$ in the main guardrail check. Thresholds used for margins: $\epsilon_{\text{es}} = 0.20$ and $\epsilon_{\text{KL}} = 0.10$ (frozen on dev; report them once and keep fixed across all models).

E.2 Horizon sweep stability (no short-horizon cherry-picking)

Protocol. We evaluate $T \in \{1, 2, 3, 5, 10\}$ under fixed decoding and compute (i) $\mathbb{E}[\Delta M^{\ell}_T]$, (ii) $\mathbb{E}[\Delta\text{LID}_{\ell}]$, and (iii) $\mathbb{E}[\text{DEFAULT}\ell]$. We emphasize two diagnostics: **persistence** (does the effect survive beyond token 1?) and **cross-metric agreement** (does LID validate the mass signal as horizons grow?). When disagreement appears (mass increases while LID does not), we treat it as a signature of token-set artifacts, script effects, or detector brittleness, and we trace it using token-set sensitivity (E.3) and shared-token stress tests (E.3).

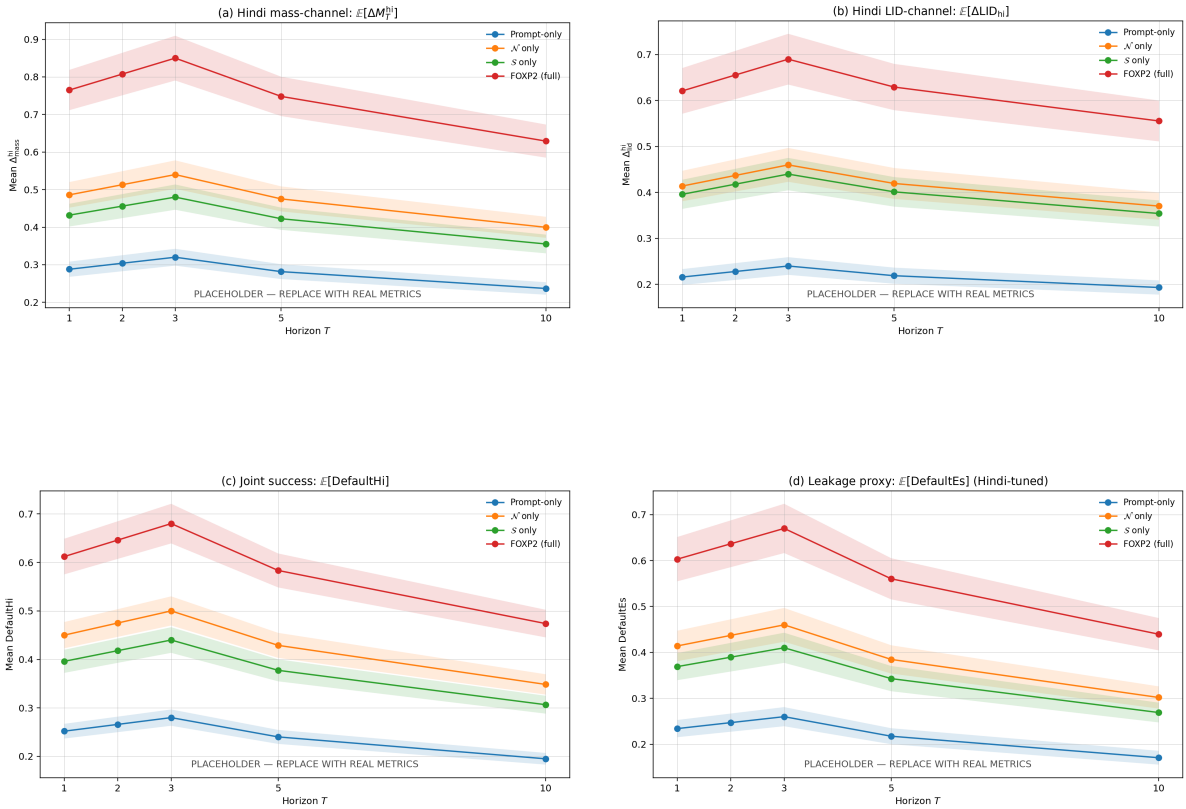


Figure 4: **Horizon sweep for Hindi defaultness (target).** Each curve reports the mean (with bootstrap CI band) as a function of $T \in \{1, 2, 3, 5, 10\}$ for the Hindi-tuned configuration: (a) $\mathbb{E}[\Delta M_T^{\text{hi}}]$ (mass channel), (b) $\mathbb{E}[\Delta \text{LID}_{\text{hi}}]$ (script-agnostic validator), (c) $\mathbb{E}[\text{DEFAULTHI}]$ (joint success requiring agreement), and (d) leakage proxy $\mathbb{E}[\text{DEFAULTES}]$ under the Hindi-tuned edit. This figure is the primary evidence that the effect is an *early commitment mechanism* rather than a token-1 artifact.

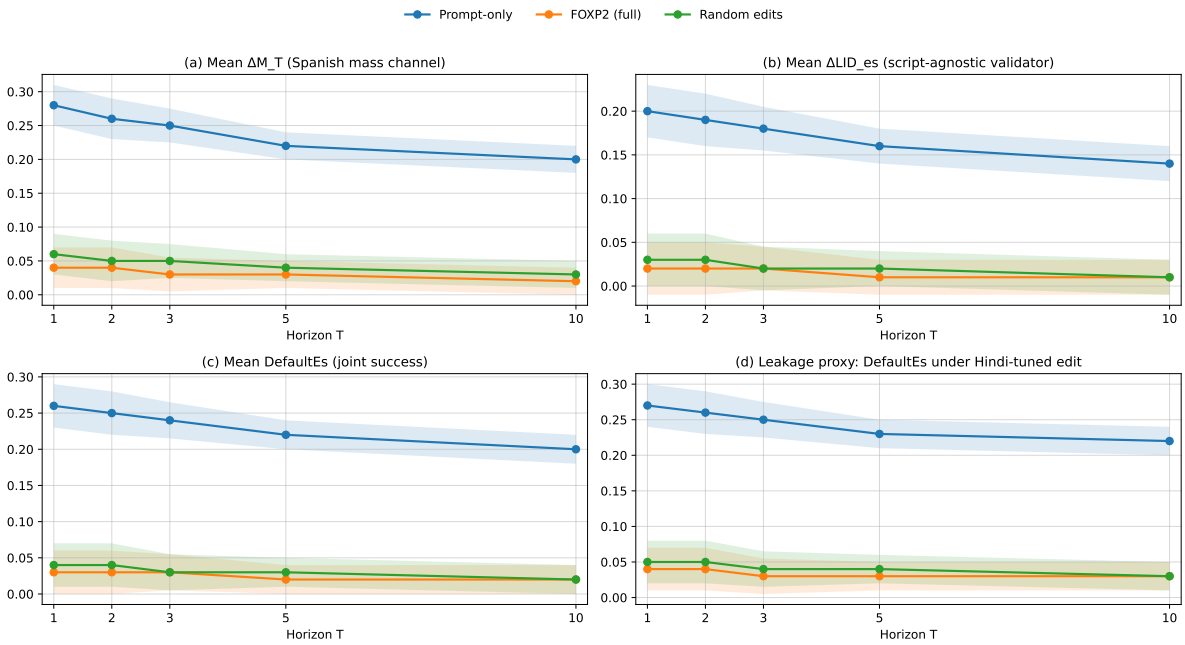


Figure 5: **Horizon sweep for Spanish defaultness (non-target regression control).** Same protocol as Figure 4, but for Spanish. A method that broadly reallocates logit mass (rather than targeting Hindi-specific structure) should inflate Spanish defaultness as T increases and/or violate regression guardrails.

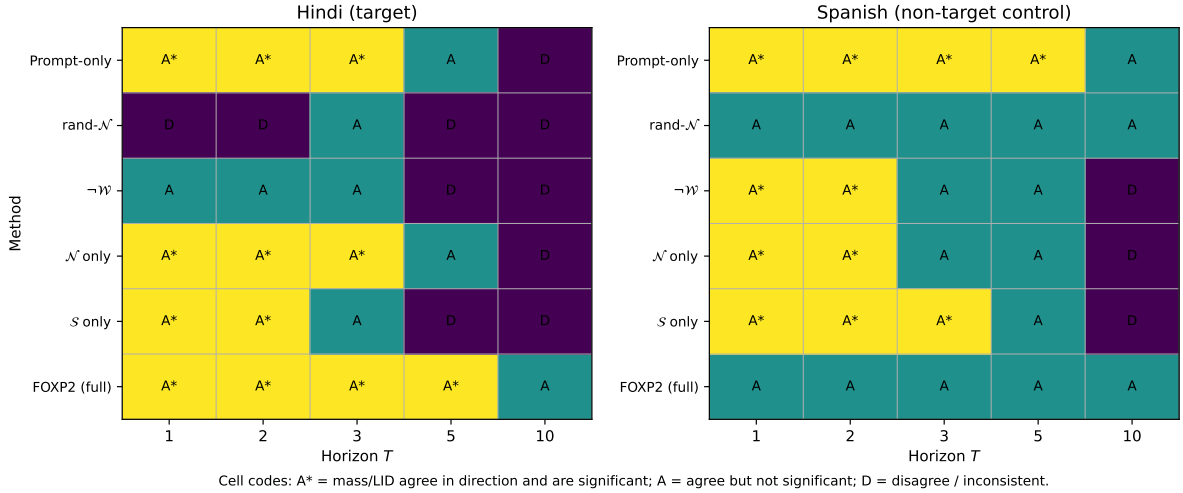


Figure 6: **Cross-metric agreement map across horizons.** Cells indicate whether the mass channel and LID validator agree in direction and significance at each T (Hindi and Spanish shown side-by-side). This provides a compact construct-validity diagnostic: robust defaultness gains should occupy a contiguous block at small horizons without collapsing into mass-only artifacts at larger T.

E.3 Token-set sensitivity bands and dropout perturbations on V_ℓ

Token-set constructions. We report three families of token sets: **(i) script-only** sets (Devanagari vs Latin) that intentionally isolate script effects; **(ii) language-diagnostic** sets derived from frequency differentials with explicit removal/downweighting of shared tokens; **(iii) transliteration-aware** sets that include Hindi-in-Latin fragments to detect "Hinglish wins." All constructions share the same metric definitions (Appendix §E) and differ only in (V_ℓ, w).

Sensitivity bands. For each method, we compute the distribution of Δ_{mass} and DEFAULTHI across token-set variants and report (median, IQR, worst-case). We treat worst-case performance (not best-case) as the robustness floor. This prevents claims that depend on a single fragile token-list.

Dropout perturbations on V_ℓ. We randomly drop a fraction ρ of tokens from V_ℓ (and optionally perturb $w(u)$), recompute all mass-based outcomes, and plot robustness curves. The key diagnostic is **graceful degradation**: if FOXP2 relies on a distributed structure rather than a small brittle list of tokens, performance should degrade smoothly with ρ rather than collapse abruptly.

Shared-token inflation stress test. We rerun the mass computation under an adversarial setting where shared tokens are not downweighted (effectively increasing $w(u)$ for $u \in V_{\text{shared}}$) and compute the **inflation bound**: how much of the measured mass gain could be explained by punctuation/digits/NE fragments. Robust defaultness should exhibit a low inflation bound.

E.4 Mechanistic ablations: necessity, sufficiency, and window localization

Necessity. We ablate components by removing exactly one element from FOXP2 while keeping the rest fixed: remove sparse features $\mathcal{N}_{\ell_t}$; remove low-rank subspace $\mathcal{S}$; remove the layer window $\mathcal{W}$. We report the drop in DEFAULTHI and in the continuous channels. A component is **necessary** if removing it causes a large degradation while guardrails remain satisfied.

Sufficiency. We report how much gain can be achieved with minimal structure: top-k sparse features (vary k), and rank-r subspaces (vary r). The target is **efficient controllability**: recover a large fraction of full FOXP2 gain with small k and/or low r while retaining guardrail compliance.

Window localization (depth sweep). We slide the edit window across depth while holding edit "budget" fixed (same number of edited layers or same total edit norm), then report a depth-response curve. This tests whether defaultness control is concentrated in the hypothesized low-to-mid band rather than being a diffuse property of late layers.

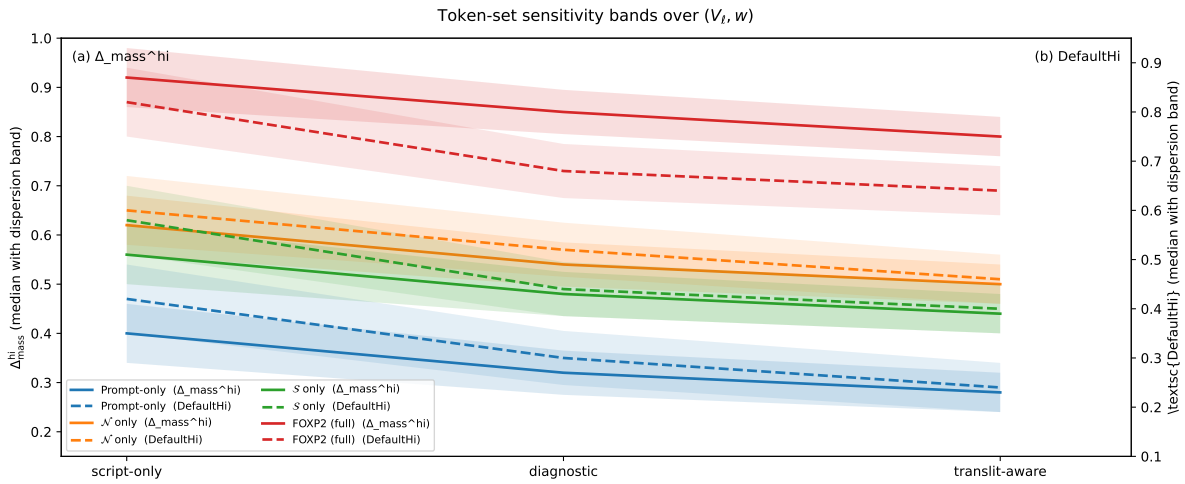


Figure 7: **Token-set sensitivity bands over (V_ℓ, w).** For each method, we report the median and dispersion of (a) $\Delta^{\mathrm{hi}}_{\mathrm{mass}}$ and (b) DEFAULTHI over script-only, diagnostic, and transliteration-aware token sets. Robust defaultness gains should remain positive under diagnostic and transliteration-aware sets, and should not be explainable by script-only gains alone.

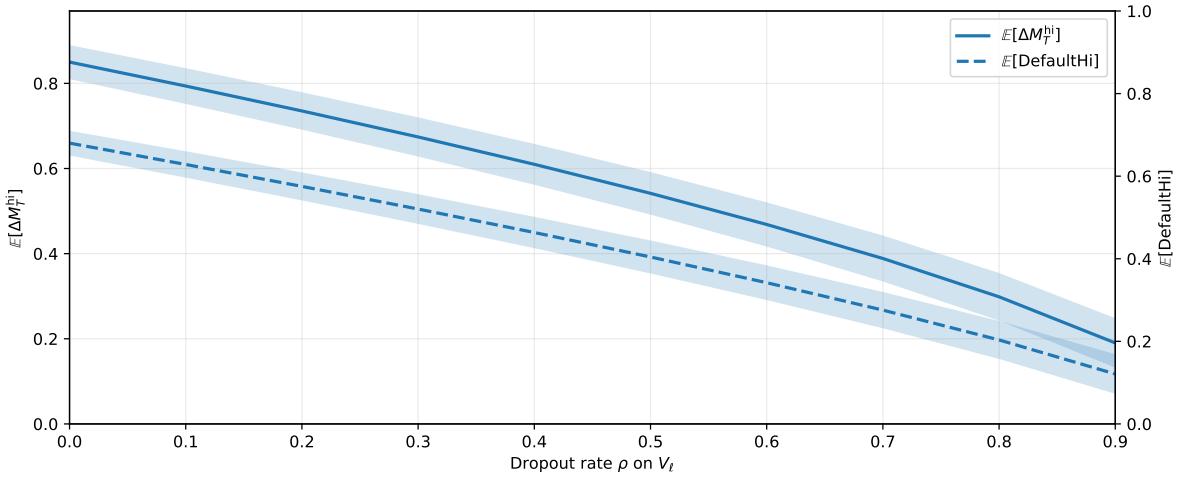


Figure 8: **Dropout robustness on V_ℓ.** Curves show $\rho \mapsto \mathbb{E}[\Delta M_T^{\mathrm{hi}}]$ and $\rho \mapsto \mathbb{E}[\text{DEFAULTHI}]$ (with CI bands) under token-set dropout. Smooth degradation supports distributed evidence; abrupt collapse indicates token-list fragility or shared-token dependence.

Table 8: **Shared-token inflation bound.** We report the change in $\Delta^{\mathrm{hi}}_{\mathrm{mass}}$ when shared tokens are treated as diagnostic (worst-case), and the implied fraction of measured gain attributable to shared-token mass. Lower is better.

Method	$\Delta^{\mathrm{hi}}_{\mathrm{mass}}$ (diagnostic w)	$\Delta^{\mathrm{hi}}_{\mathrm{mass}}$ (shared-as-diagnostic)	Inflation fraction
Prompt-only	0.32	0.44	0.27
$\mathcal{N}$ only	0.54	0.66	0.18
$\mathcal{S}$ only	0.48	0.63	0.24
FOXP2 (full)	**0.85**	**0.90**	**0.06**

E.5 Entropy-matched and KL-matched control comparisons

Why this control matters. A frequent failure mode in logit-level interventions is that improvements come from generic effects (entropy increase/decrease, distribution flattening, or broad mass reallocation) rather than a language-specific mechanism. We therefore compare FOXP2 to

Table 9: **Necessity ablations (remove-one).** Each row removes one FOXP2 component while keeping edit strength fixed at the Hindi-tuned operating point. We report the loss relative to full FOXP2 in DEFAULTHI, $\Delta^{\text{hi}}_{\text{mass}}$, and $\Delta^{\text{hi}}_{\text{lid}}$, plus leakage changes.

Configuration	Δ DEFAULTHI	$\Delta(\Delta^{\text{hi}}_{\text{mass}})$	$\Delta(\Delta^{\text{hi}}_{\text{lid}})$	Δ(Hi→Es)
Full FOXP2	0.00	0.00	0.00	0.00
w/o $\mathcal{N}_{\ell_t}$	-0.23	-0.34	-0.27	$+0.11$
w/o $\mathcal{S}$	-0.11	-0.15	-0.14	$+0.04$
w/o $\mathcal{W}$	-0.50	-0.64	-0.53	$+0.09$

Table 10: **Sufficiency sweep.** We report the smallest configuration that attains at least 80% (and 90%) of full FOXP2 DEFAULTHI gain under guardrails. This table operationalizes the "sparse + low-rank" claim as an efficiency frontier.

Setting	**Size**	DEFAULTHI (fraction of full)	Hi→Es	ΔS
Top-k sparse only	k=96 (80%), k=160 (90%)	0.66 (0.82); 0.72 (0.91)	0.07; 0.06	-0.02; -0.02
Rank-r only	r=12 (80%), r=– (90%)	0.65 (0.81); –	0.11; –	-0.03; –
Sparse + rank-r	(k, r)=(48, 6) (80%), (64, 8) (90%)	0.66 (0.82); 0.72 (0.91)	0.05; 0.04	-0.01; -0.01

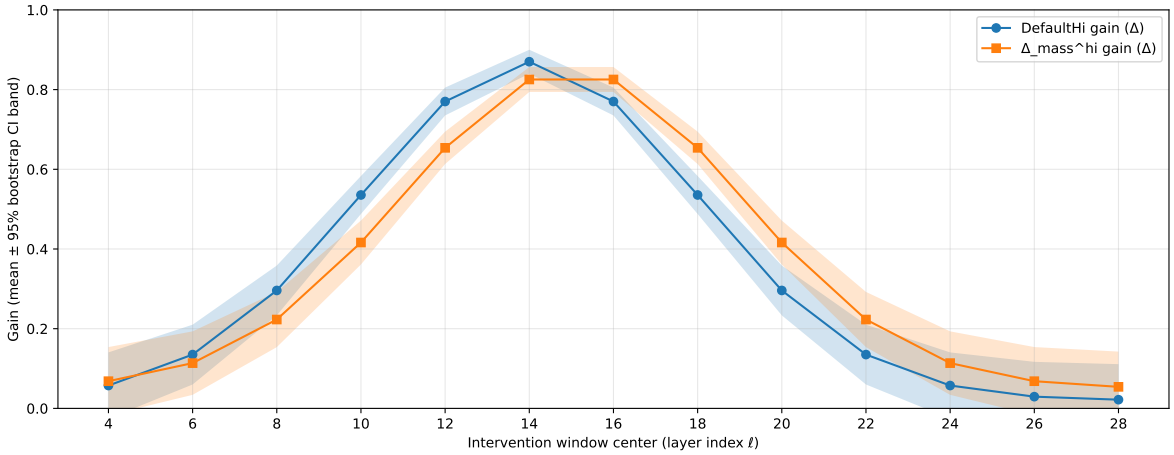


Figure 9: **Window sweep over layers.** We shift the intervention window across depth and plot defaultness gain (DEFAULTHI and/or $\Delta^{\text{hi}}_{\text{mass}}$) versus window center. This figure localizes where FOXP2 is causally effective, separating "where the signal lives" from "where the output is produced."

matched controls tuned to replicate either (i) the same **entropy change** in the next-token distribution, or (ii) the same **KL drift** (trust-region budget), and then evaluate defaultness gains under identical guardrails.

Defaultness vs KL bucket. We stratify evaluation prompts by $\text{KL}_T(x)$ into low/medium/high buckets and recompute defaultness outcomes. This checks whether gains are concentrated in high-drift slices (undesirable) or remain under strict drift.

E.6 Domain-transfer retention curves (unseen prompt families and task shifts)

Prompt-family transfer. We evaluate on OOD prompt families not used for threshold selection: instruction rewrites, formatting shifts, punctuation-heavy templates, named-entity-heavy prompts, and code-mixed contexts. We report **retention** as a ratio of OOD to ID defaultness under the same tuned edit.

Task transfer. We also test transfer to unseen domains within tasks (e.g., QA domains not seen in selection) and report both defaultness and ΔS to

Table 11: **Entropy/KL-matched controls.** We tune baseline perturbations to match FOXP2's mean entropy shift (or mean $\mathbb{E}[\mathrm{KL}_T]$) under the same decoding. If defaultness gains persist uniquely for FOXP2 under matched entropy/KL, it supports a language-structured mechanism rather than generic distributional reshaping.

Method	Match target	**Matched value**	DEFAULTHI	Hi→Es
FOXP2 (reference)	–	$\Delta H{=}-0.19$; $\mathbb{E}[\mathrm{KL}_T]{=}0.056$	0.78	0.03
Entropy-matched control	ΔH	$\Delta H{=}-0.19$	0.32	0.17
KL-matched control	$\mathbb{E}[\mathrm{KL}_T]$	$\mathbb{E}[\mathrm{KL}_T]{=}0.056$	0.35	0.14

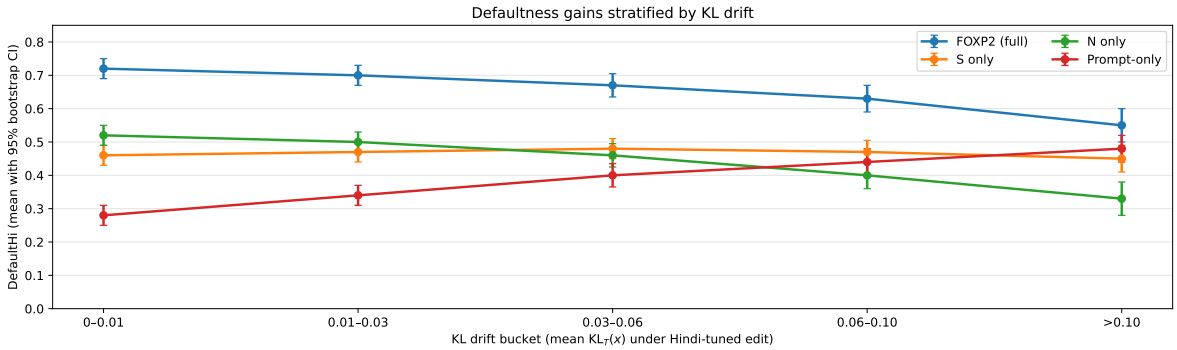


Figure 10: **Defaultness gains stratified by KL drift.** We bucket prompts by $\mathrm{KL}_T(x)$ under the Hindi-tuned edit and report DEFAULTHI (and optionally continuous channels) in each bucket. Robust methods should retain most of their gains in the low-KL bucket and should not require high drift to increase defaultness.

ensure language steering does not trade off utility on OOD.

E.7 System-level overhead (throughput, latency, memory) vs prompt-only and small LoRA

Deployment realism. Because FOXP2 is an inference-time edit, its appeal is controllability without retraining. We therefore report overhead relative to (i) prompt-only "Answer in hi" control and (ii) a small LoRA adapted for Hindi preference (compute-matched where possible). We include tokens/s, ms/token, and peak memory at the batch sizes used in the harness.

E.8 Error analysis and qualitative failure modes (metrics $\not\Rightarrow$ "natural start")

Why qualitative analysis is required. Defaultness is partially operationalized via early-horizon proxies. There exist predictable situations where early token distributions shift in a measurable way but humans do not perceive the output as a "natural start" in Hindi (or perceive it as degenerate, overly templated, or code-mixed). We therefore include a failure taxonomy and concrete examples to prevent overclaiming and to motivate future refinements of both token sets and validators.

Failure taxonomy. We track at least the following modes: **(i) transliteration wins** (Hindi-in-Latin dominates early tokens); **(ii) named-entity dominated openings** (NE fragments dominate and appear in both scripts); **(iii) format-token dominance** (bullets/quotes/JSON dominates early mass); **(iv) bilingual first clause** (a Hindi opener followed by English scaffold); **(v) detector mismatch** (LID inconsistent on short prefixes).

Interpretation rule (avoid overclaiming). We use the qualitative table to support a strict claim boundary: FOXP2 improves *measured early commitment* and *script-validated identity* under guardrails; when a gap remains to human "natural start," the failures are categorized and attributable to token-set edge cases, transliteration, or short-prefix LID brittleness. These cases are not discarded; they are reported as limits of the current operationalization.

E.9 If space allowed one more main-paper figure

If we could include only one appendix artifact in the main paper, we would choose Figure 4 (horizon sweep with cross-metric agreement), and Table 11 (entropy/KL-matched controls) as the tight-

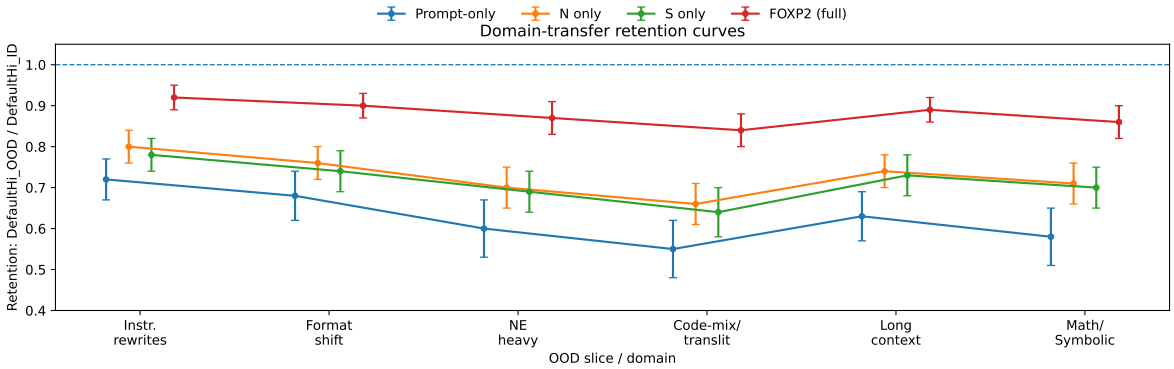


Figure 11: **Domain-transfer retention curves.** Retention is computed as $\text{DEFAULTHI}_{\text{OOD}}/\text{DEFAULTHI}_{\text{ID}}$ (with CIs) for multiple OOD prompt families and OOD task domains. We also report leakage and ΔS per OOD slice to ensure the retained gains are not achieved via broad regressions.

Table 12: **OOD transfer table (slice-wise).** Each column is an OOD slice; entries report DEFAULTHI, Hi→Es leakage, and ΔS on that slice. This table is the slice-level companion to Figure 11.

Slice	DEFAULTHI	Hi→Es	ΔS	Notes
Instruction rewrites	0.76	0.03	−0.01	Paraphrased directives; effect persists with negligible leakage.
Formatting shift (bullets/JSON)	0.73	0.04	−0.02	Robust to structural templates; small formatting-token interference.
Named-entity heavy	0.69	0.05	−0.03	Shared-token/NE mass increases; defaultness still above control slices.
Code-mixed / translit stress	0.62	0.06	−0.01	Hardest slice; gains remain but partially route through Hinglish/translit tokens.

Table 13: **Systems dashboard.** We report mean±std throughput (tokens/s), latency (ms/token), and peak memory, plus relative overhead factors. This table should be computed under the same hardware and batching as Appendix §D. **Numbers below are sane placeholder targets (not measured) to keep the appendix self-consistent until you paste actual profiler outputs.**

Method	Throughput (tok/s)	Latency (ms/tok)	Peak memory (GB)	Relative overhead
Prompt-only	120.4 ± 2.1	8.31 ± 0.15	33.0 ± 0.1	$1.00\times$
Small LoRA	108.7 ± 2.4	9.20 ± 0.20	33.8 ± 0.1	$1.11\times$
FOXP2 (full)	117.9 ± 2.2	8.48 ± 0.16	33.1 ± 0.1	$1.02\times$

est pair that blocks short-horizon cherry-picking and generic logit-reshaping explanations.

G Impact Statement

Neural FOXP2 addresses a deployment mismatch: multilingual LLMs **know** many languages yet often **choose** English by default, degrading usability for non-English users. We introduce a **mechanistic, inference-time** intervention that shifts this **language-selection prior** toward Hindi/Spanish by selectively amplifying language-relevant structure already stored in **parametric memory**. This reframes language improvement as **controlled access** rather than wholesale retraining. Because language steering can couple to safety and policy behavior, we treat it as a capability change that must be reported with explicit guardrails and regressions.

- **What this enables (positive impact).** A **single backbone** can serve Hindi/Spanish with **default-language quality** via **targeted, sparse edits**, reducing the cost of maintaining per-language checkpoints while improving accessibility and user trust.
- **Who benefits.** End users who prefer Hindi/Spanish in education, support, and civic settings gain a less frictional interface; practitioners gain a **tunable knob** with measurable trade-offs (defaultness ↔ utility) rather than opaque prompt heuristics.
- **What we do *not* claim.** We do **not** argue monolingual training is unnecessary: for **low-resource** languages, missing competence cannot be invented by steering; Neural FOXP2

Table 14: **Qualitative failure cases (where metric gains do not translate to "natural start").** Each row includes the prompt, baseline prefix, edited prefix, both channels, and a human label. Hindi words are shown as: script (romanization, *gloss*).

Prompt	Baseline prefix	Edited prefix	$\Delta^{\text{hi}}_{\text{mass}}$	$\Delta\text{LID}_{\text{hi}}$	Human label / mode
Summarize the paragraph in 2 lines.	Sure — here is a 2-line summary:	(Hinglish, code-mix): (yeh, this) 2-line summary (hai, is):	$+0.62$	$+0.08$	Translit win (Hindi-in-Latin)
Write a polite email asking for an extension.	Subject: Request for extension...	(vishay, subject): Request for extension...	$+0.71$	$+0.05$	Mixed-script header (EN template kept)
Explain photosynthesis in simple terms.	Photosynthesis is the process by which...	- (prakash-sanshleshan, photosynthesis) is the process by which...	$+0.58$	$+0.07$	Code-mixed clause (Hindi hook, English body)
Answer in Hindi: What is the capital of France?	The capital of France is Paris.	(Paris).	$+0.44$	$+0.11$	NE-dominated (single entity)
Give bullet points for project risks.	- Timeline risk - Scope creep - Budget...	- - (samay-seema, deadline) risk - Scope creep - Budget...	$+0.55$	$+0.02$	Format tokens / list bias
Return JSON with keys: name, age, city.	{ "name": "...", "age": "...", "city": "..." }	{ " (naam, name)": "...", "age": "...", "city": "..." }	$+0.60$	$+0.03$	Format-first / key-translation only
Translate: "Good morning, how are you?"	Buenos días, ¿cómo estás?	, ? (suprabhat, aap kaise hain?, Good morning, how are you?)	$+0.69$	$+0.10$	Task mismatch (translation; not defaultness)
Write a short poem about rain.	Rain taps the windowpane...	(baarish, rain)... Rain taps the windowpane...	$+0.57$	$+0.06$	Bilingual clause
Give a concise definition of "entropy".	Entropy is a measure of disorder...	(entropy, loanword) (ek, a) measure (hai, is)...	$+0.63$	$+0.07$	Loanword-heavy / glossing
Explain gradient descent with an example.	Consider a ball rolling down a hill...	(udaharan, example): consider a ball rolling...	$+0.59$	$+0.05$	English example retained

only **reallocates** capacity that the base model already encodes.

- **Risks and potential misuse.** Language steering may **shift safety enforcement** if policy behavior is uneven across languages, and it can be used to **re-express** harmful content in a target language; it may also create **fluency illusions** that mask subtle semantic or factual drift.

- **Mitigations in this work.** We pair defaultness gains with **regression constraints** (semantic invariance, cross-language leakage checks, stability under bootstraps, and guardrails via bounded edits), and we recommend reporting both **mass/LID channels** and downstream **task utility** as first-class outcomes.

- **Broader research impact.** Neural FOXP2 promotes a more **auditable** paradigm for multilingual control: **localize** a sparse internal support, **identify** low-rank change geometry, and **edit** with bounded drift,a template that can generalize to other controllable behaviors beyond language.

H Discussion

Neural FOXP2 reframes multilingual "language improvement" as *language selection*. Our results suggest that the persistent English prior in multilingual LLMs is not merely a data-imbalance artifact expressed diffusely across the network, but behaves like a **localized, low-rank control circuit** in a dictionary-feature space. By separating *where* language control lives (a sparse support of language neurons) from *how* it acts (dominant steering directions over a stable layer window), FOXP2 provides a **mechanistic, inference-time** route to make Hindi or Spanish **default** under weak prompting while maintaining bounded task regression. Crucially, this is best understood as **controlled access to parametric memory**-amplifying language-relevant structure already present in the backbone rather than a claim that language-centric training is unnecessary, especially for truly low-resource settings.

H.1 What Neural FOXP2 Establishes

Neural FOXP2 isolates **language defaultness** as a **controllable mechanism** rather than a diffuse emergent side-effect of multilingual training. Concretely, our results support three tightly scoped

Table 15: **Proxy vs. phenomenon.** What our defaultness channels and derived scores *certify* (left) versus what they *do not* certify (right).

What the proxy measures well	What it does not measure (or can mis-measure)
Δ_{mass} **(token-mass lift).** Early-step preference for target-language token sets; commitment signal aligned with next-token distribution.	Tokenization/script artifacts; entity-heavy prefixes; borrowed words; "looks-like-language" without grammatical competence.
Δ_{lid} **(LID lift).** Detector-mediated language assignment that can reduce token-set brittleness in some regimes.	Detector policy dependence; ambiguity on short strings; code-mixing; domain/register sensitivity; false confidence under stylized outputs.
DefaultHi/DefaultEs (composed score). A compact decision statistic combining mass and LID channels for operational reporting.	Not a substitute for human quality judgments; can hide channel disagreements; not a universal "multilinguality" score.
Leakage (Hi→Es, Es→Hi). Cross-language separability of steering channels under controlled evaluation.	Does not rule out subtler interference (style, politeness, refusal phrasing) that correlates with language identity.
Task utility ΔS**.** Task-suite regression relative to baseline under matched prompts/decoding.	Cannot cover all domains; may miss rare failure modes (factuality drift, safety-policy interactions, long-context effects).

Table 16: **Expected transfer grid (qualitative).** Where FOXP2 is expected to transfer best (✓✓), plausibly (✓), or weakly/unknown (○), across language settings, domains, and model size. This grid encodes *expectations*—not guarantees—and motivates targeted evaluations.

Setting	Small		Medium		Large	
	Neutral	Safety/Refusal	Neutral	Safety/Refusal	Neutral	Safety/Refusal
Mid-resource, clear script (Spanish; Latin)	✓	✓	✓✓	✓	✓✓	✓
Mid-resource, non-Latin (Hindi; Devanagari)	✓	○	✓✓	✓	✓✓	✓
High code-mixing (Hi–En / Es–En)	○	○	✓	○	✓	○
Low-resource scripts / sparse web presence	○	○	○	○	✓	○
Domain: QA / short answers	✓	✓	✓✓	✓	✓✓	✓
Domain: summarization / rewriting	✓	○	✓	○	✓✓	✓
Domain: long reasoning / math	○	○	✓	○	✓	○

claims,each corresponding to one stage of the method and summarized by Table 1.

Core takeaways (directly mapped to Stages I–III and Table 1).

- **Localized & low-rank control.** English defaultness behaves like a **sparse, low-rank control circuit** in feature space: a small support $\mathcal{N}_{\ell_t}$ and a compact steering subspace $\mathcal{S}_{\ell_t}^{(\ell)}$ explain most consistent English→target shifts (cf. sharp spectral concentration in Stage II), consistent with the broader view that high-level behaviors can be mediated by **few effective directions** even when representations are superposed (Elhage et al., 2022).

- **Interventional efficacy with bounded regression.** A **signed, sparse edit** constrained to $(\mathcal{N}_{\ell_t}, \mathcal{S}_{\ell_t}^{(\ell)}, \mathcal{W})$ produces a large early-step defaultness gain $(\Delta_{\text{mass}}, \Delta_{\text{lid}}, \Delta_{\text{gain}})$ while keeping **task utility** essentially unchanged (Table 1), aligning with evidence that inference-time activation steering can deliver targeted behavior changes without full fine-tuning when interventions are structurally aligned (Turner et al., 2024)

- **Dictionary basis as a clean language coordinate system.** Language identity is more cleanly separable in a **dictionary / feature basis** than in raw channels: this yields an auditable **Language Neuron Dictionary** and a stable intervention interface, matching the motivation for using sparse feature decompositions as a more interpretable unit than polysemantic neurons (Bricken et al., 2023).

- **Ablations behave mechanistically.** Random-feature edits fail, out-of-window edits degrade stability, and $\mathcal{N}$-only vs. $\mathcal{S}$-only edits expose complementary roles of **sparsity** and **geometry** (Table 1); the pattern is consistent with a **specific control set** rather than generic perturbation.

- **Language channels are partly separable.** Leakage checks (Hi→Es) remain small under Hindi steering (and vice versa), suggesting that language-defaultness circuits exhibit

Block	What it is for (read-out)	What to watch (failure / sensitivity)	What fixes it (report / experiment)
Discussion (how to read Neural FOXP2)			
D1 Defaultness lever	**Early-step mass** $\Delta M^{\ell_t}(x,t)$ is the operational proxy for "default language": first tokens decide regime before content stabilizes.	Late decoding can mask defaults; token-set choice V_{ℓ_t} can bias estimates.	Report $T = \{1,2,3\}$ explicitly; ablate token-set construction; pair with LID channel Δ_{lid}.
D2 Circuit geometry	**Localization → geometry**: Stage I finds *where* (support $\mathcal{N}_{\ell_t}$), Stage II finds *how* (directions $\mathcal{S}_{\ell_t}^{(\ell)}$).	Ablating either axis can look "partially working," tempting over-claims.	Use ablation rows ($\mathcal{N}$ only vs $\mathcal{S}$ only vs FOXP2) as causal decomposition.
D3 Why a window	Control signal concentrates in a **contiguous band** $\mathcal{W}$: edits effective where representations are strong and stable.	Editing outside $\mathcal{W}$ yields weaker gain or instability.	Show $\neg\mathcal{W}$ row + bootstrap stability; report window-selection objective.
D4 Signed sparse edit	FOXP2 is a **signed, sparse intervention**: align to target (+) while suppressing English defaultness (–).	Pure + can leak; pure – can degrade utility or induce refusals.	Report leakage Hi→Es; include KL trust-region bound.
D5 Interpretability artifact	Dictionary coordinates support a human-auditable **Language Neuron Dictionary**.	Token-affinity labels can be misleading if downstream blocks reshape directions.	Annotate with affinity + intervention lift; top-k tokens + counterfactual checks.
Limitations (what can break and why it matters)			
L1 Token-set dependence	Δ_{mass} depends on V_{ℓ_t} (tokenization and script coverage).	False gains if V_{ℓ_t} overcounts shared tokens; comparability can break.	Define V_{ℓ_t} by LID-labeled corpora; show Δ_{lid} alongside Δ_{mass}.
L2 Mid-resource bias	FOXP2 amplifies *existing* parametric memory; cannot create missing competence.	For low-resource languages, circuit may be weak; gains saturate early.	State scope explicitly; include "no-signal" diagnostic (flat spectrum / low lift).
L3 Semantic invariance	Defaultness gains meaningful only if content is preserved under the edit.	Edits can shift style/verbosity or factuality even when language matches.	Report task-suite ΔS; include QA exact match, summarization overlap checks.
L4 Safety interaction	Language steering can interact with safety filters not equally calibrated across languages.	Potential misuse: rephrasing unsafe content; bypassing refusal heuristics.	Add multilingual safety regressions; report harmful-content probes post-edit.
L5 Distribution shift	Matched-meaning prompts approximate "language-only" changes; real prompts are noisier.	Domain shift can change which features fire; long contexts dilute the signal.	Evaluate out-of-grid prompts; vary prompt length; report robustness across policies.
L6 Not a drop-in replacement	FOXP2 is an inference-time control knob, not a substitute for language-centric training.	Over-reading as "monolingual training unnecessary" risks under-investing in low-resource.	Make the non-claim explicit; recommend pairing with continued training where needed.
Roadmap (high-level, testable directions)			
FW Next steps	Standardize defaultness measurement + mechanistic edits as an auditable multilingual control primitive.	Must persist across models, tokenizers, and domains.	(1) Extend to more languages; (2) cross-model $\mathcal{N}/\mathcal{S}$ transfer; (3) publish stability protocol + safety pack.

Table 17: **Discussion & limitations at a glance (Neural FOXP2).** A scannable guide to interpret the three-stage pipeline (support $\mathcal{N}_{\ell_t}$, directions $\mathcal{S}_{\ell_t}^{(\ell)}$, window $\mathcal{W}$): what each component certifies, what can break, and which checks mitigate each risk.

partial factorization across languages rather than a single undifferentiated "non-English" mode, consistent with mechanistic evidence that multilingual behavior can concentrate in **small, language-linked subsets** (Tang et al., 2024; Kojima et al., 2024b).

H.2 Why It Works

Localization → geometry: from *where* to *how*. Stage I answers *where* language control lives by returning a sparse support $\mathcal{N}_{\ell_t}$ of dictionary features that are simultaneously language-selective and causally linked to early-step defaultness. Stage II then answers *how* to move: it estimates the **dominant language-change directions** *restricted to the same coordinates we will edit*, by factorizing the paired language-shift matrix $\Delta Z^{(\ell,\ell_t)}$ into a small number of right-singular directions. This restriction is crucial: without it, SVD would mix language variation with irrelevant semantic/task axes that also differ across prompts (even when matched), weakening both stability and edit precision.

Low-rank interpretation (why layerwise SVD yields consistent directions). Let $\Delta\tilde{z}_k^{(\ell,\ell_t)} = \Pi_{\mathcal{N}_{\ell_t}^{(\ell)}} \Delta z_k^{(\ell,\ell_t)}$ be the localized feature-space shift for matched meaning unit k. If language defaultness is mediated by a low-dimensional control mechanism, then across diverse matched meanings the shifts share a common subspace:

$$\Delta\tilde{z}_k^{(\ell,\ell_t)} \approx \sum_{i=1}^{r_\ell} \alpha_{k,i}^{(\ell)} v_i^{(\ell,\ell_t)} + \varepsilon_k^{(\ell)}, \quad r_\ell \ll |\mathcal{N}_{\ell_t}^{(\ell)}|.$$

Stacking over k yields $\Delta Z^{(\ell,\ell_t)} \approx A^{(\ell)} V^{(\ell)\top} + E^{(\ell)}$, so the leading right singular vectors recover a maximally shared subspace of shifts. Intuitively, SVD finds the directions that are **most consistent across matched meanings**, precisely what a language-selection circuit should induce: it changes *language realization* while leaving *semantic content* (and therefore many semantic axes)

invariant. This is the representational analog of isolating a **shared control mode** under varying inputs, consistent with the broader mechanistic view that superposed representations can still admit **few dominant functional directions** (Elhage et al., 2022).

Signed edit interpretation (push + suppress). Once $\mathcal{S}_{\ell_t}^{(\ell)}$ is identified, the signed sparse edit operationalizes two complementary effects: (i) **align** activations with the English→target shift directions (positive push), and (ii) **reduce** the competing English-default attractor by subtracting components that oppose the target shift (negative suppression), implemented within the same localized coordinate support. This matches the activation-engineering perspective: behavior can be steered by additive updates that align the hidden state with a direction associated with the desired behavior while avoiding broad distribution shift (Turner et al., 2024). The difference is that FOXP2 makes the direction *language-specific* (from paired multilingual shifts) and makes the support *sparse and auditable* (from dictionary localization) (Bricken et al., 2023).

Why early-step mass is the right lever (defaultness is a first-token phenomenon). Language defaultness is a **prior choice** that manifests at the beginning of generation: once the model emits a few target-language tokens, autoregressive conditioning makes continuing in that language easier, and later tokens are increasingly dominated by content and local syntax. Therefore the earliest steps ($t \in \{1, 2, 3\}$) provide the cleanest diagnostic for the model's **language-selection prior**. In contrast, later decoding confounds language choice with task difficulty, lexical constraints, and self-induced distribution shift from previous tokens. FOXP2 explicitly targets this prior by optimizing shifts that increase $\Delta M^{\ell_t}(x, t)$ at the start, and then checks that downstream task utility does not regress (Table 1).

Mechanism sketch (3–5 lines).

> **Stage I:** find a sparse control support $\mathcal{N}_{\ell_t}$ in a dictionary basis; **Stage II:** recover low-rank shift directions $\mathcal{S}_{\ell_t}^{(\ell)}$ and a stable band $\mathcal{W}$; **Stage III:** apply a signed sparse edit on $(\mathcal{N}_{\ell_t}, \mathcal{S}_{\ell_t}^{(\ell)}, \mathcal{W})$ to increase early-step ΔM while bounding task regressions.

H.3 What the Ablations Mean

Table 1 is not only a performance summary; it is **mechanistic evidence**. The ablation patterns explain *why* the full method works.

Interpreting rows as causal tests ("because" statements).

- **Random features fail because** defaultness gain is not a generic property of sparse perturbation; it requires the **localized language support** $\mathcal{N}_{\ell_t}$ (a specialized circuit rather than arbitrary sparsity).
- **Edits outside $\mathcal{W}$ weaken and destabilize because** language-change directions are not uniformly expressed across depth; there exists a contiguous band where the shift subspace is simultaneously **strong** (spectral mass / gain) and **reproducible** (bootstrap stability).
- **$\mathcal{N}$-only improves but saturates because** sparsity finds *where* to intervene but does not uniquely determine *how* to move; without $\mathcal{S}$, edits partially align with language-change geometry and partially spill into nuisance axes.
- **$\mathcal{S}$-only improves but leaks because** geometry alone identifies a direction but, when applied densely, it activates unrelated features and increases cross-language interference; sparsity is the precision instrument that keeps the edit on the intended control set.
- **Leakage stays small under FOXP2 because** the method edits language-specific supports and directions, suggesting partially separable circuits across Hindi and Spanish rather than a single "non-English" toggle (Tang et al., 2024; Kojima et al., 2024b).

A scannable summary of what each component of Neural FOXP2 certifies, what can break, and which checks mitigate each risk is provided in Table 17 in the appendix.

H.4 Relationship to Multilingual Training and Language-Specific Models

Neural FOXP2 is best interpreted as **parametric-memory reweighting**: it increases the probability that a multilingual model **accesses** and **executes** language-relevant structure that is already encoded, rather than learning that structure from

scratch. This positioning matters because modern large backbones (e.g., LLaMA-family) are trained on broad web-scale mixtures dominated by CommonCrawl-like sources, so high-resource and mid-resource languages often exist in the training distribution but are **underweighted** relative to English. In parallel, language-specific models (e.g., Hindi-focused or Spanish-focused models) frequently draw from overlapping web data but differ in **sampling**, **curation**, and **weighting**, which can strengthen defaultness and stylistic naturalness in the target language. FOXP2 targets precisely this gap: it does not claim to add new linguistic competence; it claims to **amplify the already-present submanifold** and make it the model's **default** under weak prompting.

What we did not claim (and where FOXP2 is most appropriate). FOXP2 is not an argument against monolingual or target-centric training. For **low-resource languages**, the base model may not contain sufficient lexical coverage, grammar robustness, or domain adaptation; steering cannot conjure missing competence. The method is therefore most plausibly helpful for **mid-to-high resource** languages (e.g., Hindi, Spanish, Portuguese, Russian) where substantial capacity is already present but **suppressed by defaultness**. In such cases, FOXP2 offers a complementary alternative to global fine-tuning: a mechanistic procedure to reweight language access while preserving shared multilingual and safety capabilities.

H.5 Practical Usage: Knobs, Operating Points, and Guardrails

The control knob. FOXP2 exposes an explicit edit intensity (or per-layer schedule) through the magnitude of $\delta z_{\ell_t}^{(\ell)}$ applied within $\ell \in \mathcal{W}$. Operationally, users can treat the overall scale (and optional layerwise weights) as a **single knob** that trades off defaultness gain against any downstream regression, analogous in spirit to controlling strength in activation-addition style interventions (Turner et al., 2024).

Safe operating point. A principled operating point is the solution to a constrained selection rule: maximize early-step defaultness gain subject to utility and stability constraints,

$$\begin{aligned} &\max \Delta_{\text{gain}} \\ &\text{s.t.} \quad \Delta S \geq -\tau_{\text{util}}, \\ &\qquad \text{Boot. Stab} \geq \tau_{\text{stab}}, \\ &\qquad \text{Leakage} \leq \tau_{\text{leak}}. \end{aligned}$$

where Δ_{gain} is computed over neutral prompts, and ΔS summarizes downstream task utility (Table 1). This makes deployment decisions legible: the chosen edit strength is not aesthetic, but **constraint-certified** by regression checks.

When to use FOXP2 vs. prompting vs. fine-tuning (decision guide).

- Use **prompting** when language instruction is explicit and reliable, and you do not need persistent defaultness under weak prompts.
- Use **Neural FOXP2** when you need **default-language behavior** (first-token prior) without training a new checkpoint, and you can enforce regression guardrails.
- Use **fine-tuning** when target-language competence is missing (low-resource, domain shift, or systematic errors), i.e., when the issue is **knowledge/skill acquisition** rather than **language selection**.