Abstract Activation Spaces for Content-Invariant Reasoning in Large Language Models

Anonymous ACL submission

Abstract

Large Language Models (LLMs) often struggle with deductive judgment in syllogistic reasoning, systematically conflating semantic plausibility with formal validity–a phenomenon known as *content effect*. This bias persists even when models generate step-wise explanations, indicating that intermediate rationales may inherit the same semantic shortcuts that affect answers. Recent approaches propose mitigating this issue by increasing inference-time structural constraints, either by encouraging abstract intermediate representations or by intervening directly in the model's internal computations; however, reliably suppressing semantic interference remains an open challenge.

To make formal deduction less sensitive to semantic content, we introduce a framework for abstraction-guided reasoning that explicitly separates structural inference from lexical semantics. We construct paired *content-laden* and *abstract* syllogisms and use the model's activations on abstract inputs to define an *abstract reasoning space*. We then learn lightweight *Abstractors* that, from content-conditioned residual-stream states, predict representations aligned with this space and integrate these predictions via multi-layer interventions during the forward pass. Using cross-lingual transfer as a test bed, we show that abstraction-aligned steering reduces content-driven errors and improves validity-sensitive performance. Our results position activation-level abstraction as a scalable mechanism for enhancing the robustness of formal reasoning in LLMs against semantic interference [1].

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities across a wide range of complex reasoning tasks. Yet, they frequently privilege semantic intuition over formal logic, systematically struggling to disentangle the *form* of an argument from its *content* (Eisape et al., 2024; Dasgupta et al., 2024). This limitation becomes particularly evident in syllogistic reasoning, a classical testbed for deductive competence, where models exhibit *content effect*: a well-documented phenomenon in human cognition whereby the perceived plausibility of a conclusion overrides the logical validity of the premises.

The conflict between semantic heuristics and logical rigour is not merely an occasional error, but a structural failure mode. Consider the contrasting syllogistic cases:

Valid Implausible
Premise 1: All things that have fins live in the desert.
Premise 2: Dolphins have fins.
Conclusion: Therefore, dolphins live in the desert.

Invalid Plausible
Premise 1: All flowers need water.
Premise 2: Roses need water.
Conclusion: Therefore, roses are flowers.

In *Valid–Implausible* arguments, the conclusion follows deductively from the premises but conflicts with world knowledge; in *Invalid–Plausible* arguments, the conclusion is factually acceptable yet unsupported by the premises. In both cases, LLMs tend to align their judgements with semantic plausibility, often misclassifying valid arguments as invalid and accepting invalid ones as valid (Valentino et al., 2025). These behaviours indicate that models implicitly introduce semantic constraints into formal deduction, effectively fabricating logical flaws when conclusions contradict prior knowledge.

Existing strategies for mitigating content effect expose a trade-off between cost, modularity, and reliability. Supervised fine-tuning can improve accuracy but is computationally expensive and prone to reinforcing superficial correlations instead of inducing stable logical abstractions (Bertolazzi et al., 2024) while CoT-based approaches offers an affordable alternative, yet often reproduces the same

[1] **code & data:** anonymous github link

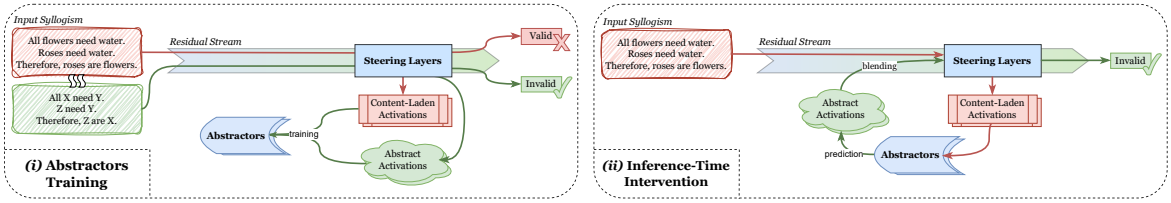


Figure 1: Overview of the abstraction steering framework. *(i)* The model processes both content-laden ("All **flowers** need **water** ...") and abstract ("All **X** need **Y** ...") syllogisms. Abstractors learn to map content-laden activations into the abstract reasoning space, *(ii)* integrating predicted targets via multi-layer activation at inference-time.

semantic intuitions within intermediate reasoning steps, failing to override implausible but deductively valid conclusions (Wan et al., 2025). Moreover, these approaches guarantees a separation between logical structure and semantic content at the level of internal representations.

These observations suggest that the core issue lies not in the representational capacity of LLMs, but in how semantic and structural signals are routed during inference. Motivated by insights from mechanistic interpretability (Kim et al., 2025), we investigate **activation steering** as an inference-time intervention (Rimsky et al., 2024; Turner et al., 2024; Lucchetti and Guha, 2025). By selectively intervening on the residual stream, activation steering enables the attenuation of semantic content signals while amplifying representations associated with formal structure.

We introduce **Abstractors**, lightweight Multi-Layer Perceptrons trained to map content-laden activations (last token activation of input "All **flowers** need **water**") onto an *abstract reasoning manifold* ("All **X** need **Y**"), dynamically predicting the target activation for each input, unlike static steering vectors (see Figure 1). Furthermore, we evaluate the zero-shot, cross-lingual transfer of this manifold by training Abstractors exclusively on English data and evaluating them on nine other languages, both high- and low-resource ones. In particular, to assess the generality of the proposed approach, we evaluate cross-lingual transfer by training Abstractors on English and testing them on nine additional languages, spanning both high-resource languages, such as Chinese and lower-resource languages, such as Telugu. This setting allows us to probe whether abstraction-aligned steering operates independently of linguistic surface form.

Overall, our contributions are as follows:

- We achieve validity-sensitive performance comparable to, and in several cases exceeding, state-of-the-art parameter-efficient methods, without modifying weights.
- We provide evidence that logical structure is encoded in a generalisable subspace, enabling inference-time transfer across unseen languages.
- Our approach offers a modular and interpretable inference-time intervention that can be enabled or disabled on demand, preserving general language-modelling capabilities while specifically targeting belief bias.

Our results position activation-level abstraction as a scalable and language-agnostic mechanism for enhancing the robustness of formal reasoning in LLMs against semantic interference.

2 Method

2.1 Overview & Problem Formulation

Let x be a syllogism in natural language. The task is to determine its formal validity $y \in \{\texttt{valid}, \texttt{invalid}\}$. We focus on intervening on the activations of the model's residual stream. We define the activation at layer ℓ and token t as $a_{\ell,t}(x) \in \mathbb{R}^d$, where d is the model's hidden dimension. Our approach involves a multi-layer intervention on a selected subset of target layers, $L^* \subset L$, typically located in the middle stage of the network, as the literature suggests that these layers encode higher-level semantic representations (Geva et al., 2021). For each layer $\ell \in L^*$, we utilise the last token activation to represent the sequence:

$$a_\ell(x) = a_{\ell,|x|}(x) \tag{1}$$

The intervention operates at inference-time without modifying the model weights. Instead of applying a static steering vector, we propose a method to dynamically predict a target representation $\hat{a}_\ell(x)$ that encodes logical structure independently of semantic content.

2.2 Abstract Reasoning Space as Target

The objective of our steering is to steer the model's activations toward representations that en-

code the logical structure of the syllogism while abstracting away semantic noise.

To define these ideal representations, we construct a parallel dataset. Each content-laden syllogism x^{con} is paired with an abstract version x^{abs} that preserves the logical form but replaces content words with abstract symbols (e.g., "All A are B"). The activations produced by the model when processing these abstract versions, $a_\ell(x^{\text{abs}})$, serve as our **target space** for pure formal reasoning. We hypothesize that mapping $a_\ell(x^{\text{con}})$ onto this "*abstract reasoning space*" reduces reliance on semantic heuristics.

2.3 Unified Activation Prediction via Contrastive Learning

At inference time, the abstract counterpart x^{abs} is not available. Therefore, we train a single Multi-Layer Perceptron (MLP) per layer, called **Abstractors**, to predict the target abstract activation $\hat{a}_\ell(x)$ directly from the content-laden activation $a_\ell(x^{\text{con}})$.

Architecture For each layer $\ell \in L^*$, we train a unified MLP f_ℓ that handles both valid and invalid syllogisms. The network employs a two-headed architecture that decouples direction and magnitude prediction:

- **Shared Backbone:** A deep feed-forward network h_ℓ extracts features from the input activation: $z = h_\ell(a_\ell(x^{\text{con}}))$
- **Direction Head:** Predicts the normalized unit vector $\hat{d}_\ell = \text{normalize}(g_d(z))$
- **Magnitude Head:** Predicts the scalar magnitude $\hat{m}_\ell = g_m(z)$

The final predicted abstract activation is:

$$\hat{a}_\ell(x) = \hat{m}_\ell \cdot \hat{d}_\ell = f_\ell(a_\ell(x^{\text{con}})) \tag{2}$$

Contrastive Training Strategy To train the unified model to correctly handle both validity classes, we employ a contrastive learning approach with triplet construction. For each training example x_i^{con}, we construct a triplet $(x_i^{\text{con}}, x_i^+, x_i^-)$ where: x_i^+ is a **positive target**: an abstract example with the same validity as x_i^{con}, x_i^- is a **negative counterfactual**: an abstract example with opposite validity.

Adaptive Matching for Triplet Construction To ensure high-quality training data, we employ a schema-based matching process. Let $\mathcal{C}^+ = \{x^{\text{abs}} \mid \text{model predicts } x^{\text{abs}} \text{ correctly}\}$ be the set of correctly-answered abstract examples. For each x_i^{con} with validity y_i:

Positive Target Selection (x_i^+):

1. **Direct Match:** If the paired $x_i^{\text{abs}} \in \mathcal{C}^+$ and has validity y_i, use it directly
2. **Schema-Based Fallback:** Otherwise, select the nearest neighbor (via cosine similarity of activations) from $\mathcal{C}^+$ with the same logical schema and validity y_i
3. **Validity-Based Fallback:** If no schema match exists, select the nearest neighbor from $\mathcal{C}^+$ with validity y_i

Negative Counterfactual Selection (x_i^-): Select the nearest neighbor from $\mathcal{C}^+$ with opposite validity $\neg y_i$, without schema constraints.

Combined Loss Function The training objective combines three losses:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{attract}} + \lambda_{\text{repel}} \cdot \mathcal{L}_{\text{repel}} + \lambda_{\text{mag}} \cdot \mathcal{L}_{\text{mag}} \tag{3}$$

where:

- **Attraction Loss** aligns the predicted direction with the positive target:

$$\mathcal{L}_{\text{attract}} = \mathbb{E}_i \left[1 - \cos(\hat{d}_\ell(x_i^{\text{con}}), d_\ell(x_i^+))\right] \tag{4}$$

  where $d_\ell(x_i^+) = \text{normalize}(a_\ell(x_i^+))$

- **Repulsion Loss** pushes the prediction away from the negative counterfactual:

$$\mathcal{L}_{\text{repel}} = \mathbb{E}_i \left[\text{ReLU}\left(\cos(\hat{d}_\ell(x_i^{\text{con}}), d_\ell(x_i^-)) - \mu\right)\right] \tag{5}$$

  where μ is a margin threshold (typically 0), and $d_\ell(x_i^-) = \text{normalize}(a_\ell(x_i^-))$

- **Magnitude Loss** matches the predicted magnitude to the positive target:

$$\mathcal{L}_{\text{mag}} = \mathbb{E}_i \left[(\hat{m}_\ell(x_i^{\text{con}}) - \|a_\ell(x_i^+)\|_2)^2\right] \tag{6}$$

This contrastive formulation enables the unified model to learn a representation space where valid and invalid syllogisms are mapped to distinct regions, without requiring explicit class labels at inference time.

2.4 Inference-Time Steering

Since the target abstraction $\hat{a}_\ell(x)$ depends on the final token's activation, we employ a **two-pass inference strategy**:

1. **Pre-computation Pass:** We perform a standard forward pass on input x to extract the unsteered activation of the final token, $a_{\ell,|x|}(x)$, and compute the target $\hat{a}_\ell(x)$ using the trained Abstractor.
2. **Steered Pass:** We re-process x. During this second pass, the pre-computed target $\hat{a}_\ell(x)$ is blended into the stream as described below.

Steering Layers We steer multiple contiguous layers rather than a single layer to ensure robust propagation of the abstract representation throughout the forward pass. Target layers L^* are selected by identifying regions where positive and negative abstract targets exhibit maximal separation (lowest cosine similarity), typically occurring in the middle layers of the network where abstract concepts have formed but remain distinct. Appendix J visualizes this layer-wise separation analysis across all models, with the selected steering layers corresponding to regions of minimal positive-negative similarity (full details in Appendix A.1).

Positional Blending The intervention intensity is modulated by a positional coefficient α_t that increases linearly along the sequence:

$$\alpha_t = \begin{cases} 0 & \text{if } t < t_{\text{start}} \\ \alpha \cdot \frac{t - t_{\text{start}}}{T - t_{\text{start}}} & \text{if } t \geq t_{\text{start}} \end{cases} \quad (7)$$

where $\alpha \in [0, 1]$ is the maximum steering strength, T is the sequence length, and t_{start} is the token position where the instruction ends and the actual syllogism content begins.

Activation Blending The steered activation for token t at layer ℓ is computed as:

$$a^{\text{steer}}_{\ell,t}(x) = (1 - \alpha_t) \cdot a_{\ell,t}(x) + \alpha_t \cdot \hat{a}_\ell(x) \quad (8)$$

Crucially, during the Steered Pass, the single target vector $\hat{a}_\ell(x)$ (computed from the Pre-computation Pass) is broadcast to all content tokens $t \geq t_{\text{start}}$ to align the entire reasoning sequence with the abstract manifold.

3 Experimental Setup

3.1 Models

We evaluate the steering pipeline on three families of open-weights LLMs. To demonstrate how our approach scales with model capabilities, we select different models: `Qwen-2.5-7B` (Qwen et al., 2024) and `3-14B` (Qwen et al., 2025); `Gemma-2-9B` (Gemma Team et al., 2024) and `3-12B` (Gemma Team et al., 2025); `Mistral-7B-v0.3` (Mistral AI, 2023) and `Ministral-3-14B` (Mistral AI, 2025)[2].

3.2 Dataset

We extend a syllogistic reasoning corpus from (Bertolazzi et al., 2024), comprising 24 logical forms instantiated via taxonomic relations from WordNet. Each instance includes two premises, a conclusion, and is annotated with logical **validity** and conclusion **plausibility**.

The primary dataset is in English (`en`). To assess cross-lingual transfer, we evaluate on nine additional languages spanning diverse families, scripts, and resource levels: **High-Resource Languages (HRLs)** (French, Spanish, Italian, German, Russian, Chinese) and **Low-Resource Languages (LRLs)** (Bengali, Swahili, Telugu). The English dataset contains 2,780 examples, each paired with its corresponding abstract counterpart, while the datasets for the other languages contain 960 examples each. All datasets are balanced across validity and plausibility categories. Data are generated via automatic translation using GPT-4o with back-translation quality control (details in Appendix B).

3.3 Metrics

Beyond simple accuracy, we introduce a suite of metrics designed to measure the quality and robustness of the model's reasoning process. These metrics quantify heuristic reliance, penalise it in a holistic score, and measure the effectiveness of steering intervention.

Belief Bias (Δ_{belief}) How much the plausibility of the conclusion influences the model's accuracy. To calculate this, we group the dataset into two categories: **Belief-Consistent**: Inputs where formal validity aligns with real-world plausibility (i.e., Valid-Plausible and Invalid-Implausible cases). **Belief-Conflict**: Inputs where logic and plausibility are at odds (i.e., Valid-Implausible and Invalid-Plausible cases).

The Belief Bias is the performance gap between these two groups, isolating the model's reliance on semantic heuristics.

$$\text{Acc}_{\text{consistent}} = \text{Acc}_{x \in \{v \iff p\}} \quad (9)$$

$$\text{Acc}_{\text{conflict}} = \text{Acc}_{x \in \{v \iff \neg p\}} \quad (10)$$

$$\Delta_{\text{belief}} = |\text{Acc}_{\text{consistent}} - \text{Acc}_{\text{conflict}}| \quad (11)$$

Bias-Penalized Accuracy (BPA) Global accuracy alone can be misleading, as a model might

[2] **code & data:** anonymous github link

achieve high performance by exploiting heuristics. We propose a holistic metric, Bias-Penalized Accuracy, that directly penalizes the model for relying on semantic content. The BPA scales the model's global accuracy by its robustness to belief bias.

$$\text{BPA} = \text{Acc}_{\text{global}} \times (1 - \Delta_{\text{belief}}) \quad (12)$$

Abstract Alignment (η) To evaluate our hypothesis that steering maps content-laden representations onto an abstract manifold, we measure how closely the steered model approaches the performance obtained on the subset of purely abstract syllogisms:

$$\eta = \frac{\text{Acc}_{\text{steered}}}{\text{Acc}_{\text{abstract}}}, \quad (13)$$

where $\eta = 1.0$ indicates the steered model matches the abstract upper bound, $\eta < 1.0$ indicates performance below the bound, and $\eta > 1.0$ indicates the steered model exceeds the abstract upper bound.

3.4 Evaluation

Cross-Validation To ensure reliable results, all Abstractors are trained and evaluated using a 3-fold stratified cross-validation scheme. The data are stratified by syllogism validity to maintain an equal class distribution within each fold. All reported results are the average across these three folds.

Cross-Lingual Transfer Our primary hypothesis is that the Abstractors learn a generalisable representation of logic across languages. To test this, Abstractors are trained exclusively on the English set, and their performance is evaluated in a zero-shot way on the test sets for English, French, Spanish, Italian, and Chinese.

Steering Strength Ablation We conduct an ablation study on the maximum steering strength hyperparameter, α, testing values from $\{0.1, 0.2, ..., 1.0\}$. The optimal α, for each model, is selected based on the highest BPA on the English validation set, and this value is used for all results. Detailed performance for each α value are shown in Appendix F.

Baselines To evaluate the effectiveness of inference-time intervention, we compare our with: Baseline (No Steering): The original model's zero-shot performance. Supervised Fine-Tuning (SFT): A parameter-efficient fine-tuning baseline. To ensure a fair comparison, we employ PiSSA (PrIncipal Singular values and Singular vectors Adaptation) (Meng et al., 2025), a state-of-the-art PEFT method that outperforms standard LoRA by initializing adapters via SVD. Implementation details are discussed in Appendix A.3. Chain-of-Thought (CoT): To evaluate whether the model's logical reasoning can be elicited purely through prompt engineering, we employ Chain-of-Thought (Wei et al., 2023). Details are discussed in Appendix A.4.

Due to computational constraints, baseline methods were evaluated on a subset of the cross-validation folds. Specifically, we trained SFT adapters for three models (`Mistral-7B`, `Qwen-2.5-7B`, and `Gemma-2-9B`) on Fold 0 only. For consistency, the same restriction was applied to CoT. Accordingly, SFT and CoT results report performance for these three models on Fold 0, while Steering results are averaged across all three folds for all models. A direct comparison with Steering evaluated on Fold 0 can be found in Appendix G.

4 Results & Analysis

4.1 Main Results: Steering Enhances Accuracy and Reduces Bias

We first evaluate the effectiveness of the Abstractors on the English test set. Table 1 shows the Bias-Penalized Accuracy (BPA) increase for all models, with Figure 2 providing a visual breakdown.

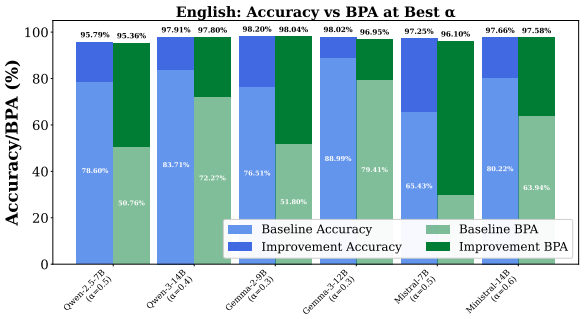


Figure 2: Performance on the English test set. Global Accuracy (blue) and Bias-Penalised Accuracy (green); darker shades indicate gains from activation steering.

The results demonstrate consistent improvements across all model families. Mapping content-laden activations towards an abstract reasoning space yields BPA gains ranging from $+16.54$ points (`Gemma-3-12B`) to $+42.99$ points (`Qwen-2.5-7B`). The weaker baseline models exhibit the most dramatic improvements, with `Mistral-7B` achieving a $+65.94$ pp increase in BPA, transforming a highly biased baseline (30.16) into a competent reasoner (96.10).

The improvements in BPA consistently exceed those in raw Global Accuracy. For instance, whilst `Qwen-2.5-7B` sees an accuracy gain of $+15.21$ pp,

Model	English						HRLs						LRLs					
	Base Model		SFT		CoT		Base Model		SFT		CoT		Base Model		SFT		CoT	
	BPA	→Steer	BPA	Δ_B	BPA	Δ_B	BPA	→Steer	BPA	Δ_B	BPA	Δ_B	BPA	→Steer	BPA	Δ_B	BPA	Δ_B
Qwen-2.5-7B	50.8	**95.4**	98.6	+3.2	81.7	-13.7	42.9	**86.1**	84.6	-1.5	57.7	-28.4	55.2	**64.8**	70.1	+5.3	55.0	-9.8
Qwen-3-14B	72.3	**97.8**	95.8	-2.0	83.1	-14.7	57.5	**86.9**	–	–	–	–	54.0	**70.8**	–	–	–	–
Gemma-2-9B	51.8	**98.0**	98.8	+0.8	69.9	-28.1	43.9	**89.0**	93.4	+4.4	58.5	-30.5	47.0	**73.5**	77.0	+3.5	53.6	-19.9
Gemma-3-12B	79.4	**97.0**	97.7	+0.7	86.8	-10.2	63.5	**90.1**	–	–	–	–	58.0	**79.0**	–	–	–	–
Mistral-7B	30.2	**96.1**	88.1	-8.0	34.3	-61.8	30.0	**75.4**	78.2	+2.8	34.2	-41.2	47.6	**53.0**	53.7	+0.7	50.0	-3.0
Ministral-14B	63.9	**97.6**	96.3	-1.3	76.2	-21.4	49.2	**86.1**	–	–	–	–	50.4	**69.2**	–	–	–	–
Average	58.1	**97.0**	95.9	-1.1	72.0	-25.0	47.8	**85.6**	85.4	-0.2	50.1	-35.5	52.0	**68.4**	66.9	-1.5	52.9	-15.5

Table 1: Comparison of Bias-Penalized Accuracy (BPA) across all methods and language groups. **Steering** (bold) shows average performance across 3 folds for all models. Δ_B shows the difference relative to Steering (green = improvement, red = degradation).

its BPA jumps by +42.99 pp. This disparity reveals that steering does not merely improve the model's thinking strategy; rather, it fundamentally alters the reasoning process by suppressing reliance on semantic content: precisely the type of heuristic dependence that BPA is designed to penalise. This pattern holds across all models: the reduction in Content Effect (reflected in the larger BPA gains) indicates that the Abstractors successfully dise3ntangle logical form from semantic content, forcing the model to reason about structure.

4.2 Cross-Lingual Generalisation

We now test a central hypothesis of this work: that the Abstractors capture a generalisable "*logic manifold*" that transcends language. To provide a structured analysis of transfer limits, we partition evaluation languages into HRLs and LRLs (as mentioned in §3.2). This stratification allows us to disentangle the effects of the steering mechanism from the quality of the base model's pre-trained representations.

HRLs: Near-Perfect Transfer Table 1 and Figure 3 reveal robust transfer across all model families. Steering achieves percentage increases in BPA that often exceed those observed in English.

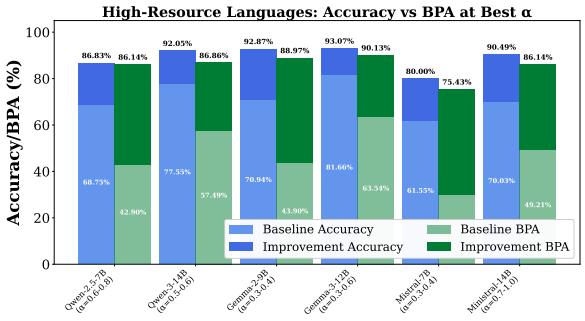


Figure 3: Performance comparison on HRLs (French, Spanish, Italian, German, Russian, Chinese).

The consistency of these results is significant. For instance, Gemma-3-12B achieves an average BPA of 90.13 across HRLs, only slightly below its English performance of 96.95. This transfer extends even to languages with fundamentally different scripts and tokenisation schemes such as Chinese, which relies on logographic characters and entirely different subword structure. This suggests the Abstractors target a deeper semantic layer encoding logical structure in a manner coherent across typologically diverse languages, rather than exploiting superficial token-level patterns.

LRLs: Meaningful but Limited Gains Whilst steering consistently improves performance, Table 1 and Figure 4 show that the absolute gains are noticeably smaller.

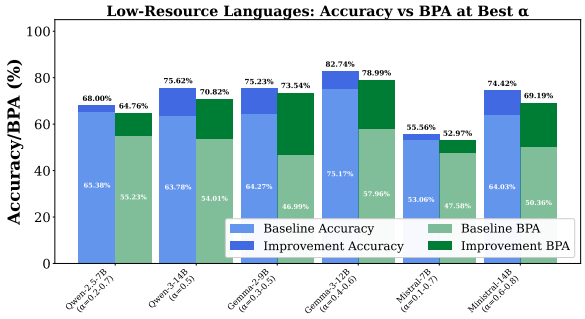


Figure 4: Performance comparison on LRLs (Bengali, Swahili, Telugu).

BPA improvements in LRLs range from +5.39 pp (Mistral-7B) to +26.55 pp (Gemma-2-9B). Whilst substantial in relative terms–often representing +30% to +50% increases–they fall short of the near-perfect alignment observed in HRLs. This gap reflects a *geometry gap* in base model representations: LRLs receive less exposure during pre-training, leading to noisier, less structured activations. Since steering amplifies and redirects

existing latent patterns rather than inducing new knowledge, it cannot produce robust logical reasoning where the underlying representation quality is poor. Still, that steering yields *any* improvement in LRLs is non-trivial. Abstractors trained without encountering any tokens of Swahili, Bengali, or Telugu still identify and enhance reasoning structures in these languages, suggesting that models maintain some cross-lingual coherence in their logical-reasoning manifolds.

Summary: The cross-lingual results reveal a clear hierarchy: *English* shows near-perfect steering; *HRLs* exhibit strong zero-shot transfer, remaining within 5–10 pp of English; *LRLs* display meaningful but constrained improvements, reflecting weaker base representations. This pattern provides compelling evidence that Abstractors learn a generalisable logical transformation.

4.3 Abstract Alignment Analysis

Beyond performance metrics, we validate the geometric mechanism of our intervention using the Abstract Alignment (η) metric, which measures how closely steered representations approach the theoretical upper bound of purely symbolic reasoning. Detailed results in Appendix D.

Model	English	HRLs	LRLs
`Qwen-2.5-7B`	1.15	1.05	0.83
`Qwen-3-14B`	1.09	1.03	0.85
`Gemma-2-9B`	1.17	1.11	0.90
`Gemma-3-12B`	1.04	0.99	0.89
`Mistral-7B`	1.14	0.96	0.65
`Ministral-14B`	1.15	1.07	0.88

Table 2: Abstract Alignment (η) across language groups. Values represent the ratio of steered accuracy to abstract upper bound accuracy at the best BPA α.

English Alignment. English models exhibit strong alignment, with η values consistently exceeding 1.0 (Table 2). This confirms that steering vectors bridge the gap between content-laden and abstract activation manifolds, improving performance by reducing noise in the Abstractor network.

High-Resource Languages (HRLs). The inference-time nature of our approach is most evident in HRLs. Despite Abstractors being trained solely on English, η scores remain clustered around 1.0 across all HRLs, demonstrating that they share an essentially isomorphic reasoning geometry with English in the model's latent space.

Low-Resource Languages (LRLs). In LRLs, alignment remains high but shows slight degradation. The variance in η scores suggests the target abstract manifold is intrinsically noisier due to limited pre-training exposure, consistent with our earlier findings on the *geometry gap* (§4.2).

4.4 Comparative Analysis

We situate Activation Steering by comparing it with Chain-of-Thought reasoning and weight-based SFT, detailed results in Appendix G. Contrary to trends in general reasoning tasks, CoT yields the lowest performance among methods and significantly underperforms steering in every scenario, with relative BPA differences ranging from -11.55% to -63.86%. SFT via PiSSA represents the upper bound of standard interventions. SFT generally performs on par with, or slightly better than, steering in `Qwen` and `Gemma`. Relative BPA differences with respect to steering only range from -7.05% to $+7.38\%$. Yet, this comes at the cost of modularity: SFT alters model weights, risking catastrophic forgetting or overfitting to the syllogistic format. Notably, for `Mistral-7B`, steering outperforms SFT, suggesting that, for some architectures, reasoning circuits are better accessed via inference-time geometry than weight modification.

Activation steering settles a unique position, balancing accuracy, control, and modularity. Unlike SFT, which operates as black-box optimisation and may reinforce superficial shortcuts, steering provides a mechanistically grounded intervention that targets the internal representation of logical validity. This results in substantial performance gains over both CoT and SFT (Appendix G, Table 13), indicating that Content Effect reflects a failure of control. Moreover, steering requires no weight updates: Abstractors are lightweight and can be toggled at inference time, preserving the model's general behaviour.

4.5 Fluency Sanity Check

A risk of training Abstractors in English is that they may implicitly bias the model toward English-centric representations, which could manifest as increased perplexity (PPL) in other languages. We measure the relative increase in perplexity across all languages on mC4 (Xue et al., 2021) as a function of steering strength α. At the optimal α identified in §4 ($\alpha \approx 0.4-0.6$), the PPL increase remains moderate ($\leq 5\%$). More significant degradation ($> 10\%$) is observed only at aggressive steering levels ($\alpha \geq 0.8$), where the abstract representations begin to surpass the syntactic features necessary

for coherent generation. PPL increases uniformly across languages, suggesting no English-specific bias and confirming that the method operates safely, correcting reasoning without altering models' generative capabilities (details in Appendix H).

4.6 OOD Analysis

We extend our evaluation to the Multilingual Massive Multitask Language Understanding (MMMLU) (Hendrycks et al., 2021) to assess how the abstraction mechanism impacts model capabilities. We report results in terms of accuracy and separately for two task categories: **Factual**, which includes subjects such as *high school geography*, *college biology*, and *high school world history*; and **Reasoning**, which includes *abstract algebra* and *formal logic*. Since our method is designed to suppress semantic content in favour of logical form, we hypothesise a "feature suppression" effect on factual tasks. Figure 5 (details Appendix I) shows the average deltas across models and languages as a function of α. It emerges that the Reasoning subset exhibits lower degradation than in Factual subjects, as expected given the mechanistic nature of the intervention: the specificity of the steering pipeline targets reasoning-related representations while inhibiting factual memory. The nature of our approach allows steering to be dynamically disabled, toggled per task, or tuned to balance reasoning enhancement with general knowledge preservation.

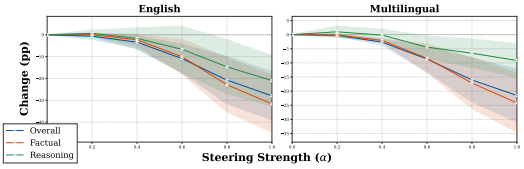


Figure 5: Performance impact on MMLU. Factual tasks (orange) and formal logic subtasks (green)

5 Related Work

Syllogistic reasoning. Syllogisms provide a controlled setting for probing whether LLMs can track logical entailment independently of lexical and world-knowledge cues (Evans et al., 1983; Newstead, 2003). Although this is a formally clear task, LLMs often fail to disentangle content from form, treating semantic plausibility as evidence of validity and exhibiting biases grounded in semantic content (Bertolazzi et al., 2024), rejecting valid conclusions when these disagree with background knowledge. Ozeki et al. (2024); Eisape et al. (2024) suggest that scaffolding paradigms yield only marginal improvements, whereas SFT reduces reliance on such heuristics but does not eliminate them.

Activation Steering. A growing body of work investigates inference-time control by intervening directly on a model's internal activations. These methods identify directions or subspaces associated with specific behaviours and modulate them to influence downstream predictions (Stoehr et al., 2024; Soo et al., 2025). Activation steering has been applied to reduce hallucinations, mitigate harmful outputs, and improve factual reliability, offering a lightweight, modular, and reversible alternative to parameter updates.

Abstraction for content-invariant reasoning. A complementary line of work improves robustness by encouraging reasoning at higher levels of abstraction. CoT explanations can inherit semantic shortcuts that compromise faithfulness (Lyu et al., 2023), motivating approaches that promote intermediate representations separating variables and relations from surface content (Ranaldi et al., 2025a). These methods rely on supervision and do not intervene on internal activations during inference.

Our contribution. We unify abstraction and activation steering by treating abstraction as an explicit geometric target for inference-time control. We define an abstract reasoning space from paired content-based and abstract syllogisms and learn lightweight *Abstractors* that map content-conditioned activations to representations aligned with this space. These predicted targets are integrated through multi-layer interventions during inference, without modifying model weights. We evaluate the resulting abstraction-aligned steering in a cross-lingual setting, assessing whether the induced transformation captures structural regularities beyond language-specific surface cues.

6 Conclusion

We showed that abstraction-guided activation steering can reduce content effect (CE) in syllogistic reasoning by separating structural inference from semantic content at the representation level. By dynamically mapping internal states to a purely logical manifold, we observe strong improvements (up to +65 points) in bias-penalized accuracy, neutralising the CE with zero-shot transfer to languages as diverse as Chinese and Bengali. Our results highlight activation-level abstraction as a scalable mechanism for improving the robustness of formal reasoning in LLMs.

Limitations

We focused on syllogistic reasoning as a controlled testbed for analysing content effects. While this setting allows for fine-grained control over logical form and semantic plausibility, it limits the generalisability of the findings to more complex reasoning scenarios, such as propositional or first-order logic, multi-hop inference, or open-ended reasoning tasks. The proposed framework relies on paired content-laden and abstract examples to define a target reasoning manifold. This design assumes that suitable abstract counterparts can be constructed reliably, an assumption that may not hold in domains where abstraction is less well-defined or inherently task-specific. Our evaluation is restricted to open-weight models, as the method requires direct access to internal activations. Thus, the approach is not directly applicable to closed-source models, in which activation-level interventions are infeasible. While this does not affect the conceptual validity of abstraction-guided steering, it constrains its practical applicability in settings dominated by proprietary systems.

References

Leonardo Bertolazzi, Albert Gatt, and Raffaella Bernardi. 2024. A systematic analysis of large language models as soft reasoners: The case of syllogistic inferences. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13882–13905, Miami, Florida, USA. Association for Computational Linguistics.

Peter Clark, Oyvind Tafjord, and Kyle Richardson. 2020. Transformers as soft reasoners over language. *Preprint*, arXiv:2002.05867.

Ishita Dasgupta, Andrew K. Lampinen, Stephanie C. Y. Chan, Hannah R. Sheahan, Antonia Creswell, Dharshan Kumaran, James L. McClelland, and Felix Hill. 2024. Language models show human-like content effects on reasoning tasks. *Preprint*, arXiv:2207.07051.

Ishita Dasgupta, Andrew Kyle Lampinen, Stephanie C. Y. Chan, Antonia Creswell, D. Kumaran, James L. McClelland, and Felix Hill. 2022. Language models show human-like content effects on reasoning. *ArXiv*, abs/2207.07051.

Boyi Deng, Yu Wan, Baosong Yang, Yidan Zhang, and Fuli Feng. 2025. Unveiling language-specific features in large language models via sparse autoencoders. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4563–4608, Vienna, Austria. Association for Computational Linguistics.

Tiwalayo Eisape, Michael Tessler, Ishita Dasgupta, Fei Sha, Sjoerd Steenkiste, and Tal Linzen. 2024. A systematic comparison of syllogistic reasoning in humans and language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8425–8444, Mexico City, Mexico. Association for Computational Linguistics.

J. St. B. T. Evans, Julie L. Barston, and Paul Pollard. 1983. On the conflict between logic and belief in syllogistic reasoning. *Memory amp; Cognition*, 11(3):295–306.

Changjiang Gao, Xu Huang, Wenhao Zhu, Shujian Huang, Lei Li, and Fei Yuan. 2025. Could thinking multilingually empower llm reasoning? *Preprint*, arXiv:2504.11833.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Gemma Team and 1 others. 2024. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. *Preprint*, arXiv:2012.14913.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Preprint*, arXiv:2009.03300.

Ruixin Hong, Hongming Zhang, Xiaoman Pan, Dong Yu, and Changshui Zhang. 2024. Abstraction-of-thought makes language models better reasoners. In *Conference on Empirical Methods in Natural Language Processing*.

Wenyue Hua, Kaijie Zhu, Lingyao Li, Lizhou Fan, Shuhang Lin, Mingyu Jin, Haochen Xue, Zelong Li, Jindong Wang, and Yongfeng Zhang. 2024. Disentangling logic: The role of context in large language model reasoning capabilities. In *Annual Meeting of the Association for Computational Linguistics*.

Geonhee Kim, Marco Valentino, and Andre Freitas. 2025. Reasoning circuits in language models: A mechanistic interpretation of syllogistic inference. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 10074–10095, Vienna, Austria. Association for Computational Linguistics.

Francesca Lucchetti and Arjun Guha. 2025. Understanding how codellms (mis)predict types with activation steering. *Preprint*, arXiv:2404.01903.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. Faithful chain-of-thought reasoning. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 305–329, Nusa Dua, Bali. Association for Computational Linguistics.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. 2025. Pissa: Principal singular values and singular vectors adaptation of large language models. *Preprint*, arXiv:2404.02948.

Mistral AI. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Mistral AI. 2025. Ministral-3 14b instruct. `https://huggingface.co/mistralai/Ministral-3-14B-Instruct-2512`. Large Language Model, Instruct-tuned.

Jacqueline L. Mitchell, Brian Hyeongseok Kim, Chenyu Zhou, and Chao Wang. 2025. Understanding formal reasoning failures in llms as abstract interpreters. *Proceedings of the 1st ACM SIGPLAN International Workshop on Language Models and Programming Languages*.

Stephen E Newstead. 2003. Can natural language semantics explain syllogistic reasoning? *Cognition*, 90(2):193–199.

Kentaro Ozeki, Risako Ando, Takanobu Morishita, Hirohiko Abe, Koji Mineshima, and Mitsuhiro Okada. 2024. Exploring reasoning biases in large language models through syllogism: Insights from the NeuBAROCO dataset. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 16063–16077, Bangkok, Thailand. Association for Computational Linguistics.

David Pomerenke, Jonas Nothnagel, and Simon Ostermann. 2025. The ai language proficiency monitor – tracking the progress of llms on multilingual benchmarks. *Preprint*, arXiv:2507.08538.

Libo Qin, Qiguang Chen, Fuxuan Wei, Shijue Huang, and Wanxiang Che. 2023. Cross-lingual prompting: Improving zero-shot chain-of-thought reasoning across languages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2695–2709, Singapore. Association for Computational Linguistics.

Qwen, :, An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, and 43 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2024. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Leonardo Ranaldi and Giulia Pucci. 2025. Multilingual reasoning via self-training. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 11566–11582, Albuquerque, New Mexico. Association for Computational Linguistics.

Leonardo Ranaldi, Giulia Pucci, Federico Ranaldi, Elena Sofia Ruzzetti, and Fabio Massimo Zanzotto. 2024. A tree-of-thoughts to broaden multi-step reasoning across languages. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1229–1241, Mexico City, Mexico. Association for Computational Linguistics.

Leonardo Ranaldi, Marco Valentino, and Andre Freitas. 2025a. Improving chain-of-thought reasoning via quasi-symbolic abstractions. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 17222–17240, Vienna, Austria. Association for Computational Linguistics.

Leonardo Ranaldi, Marco Valentino, Alexander Polonsky, and André Freitas. 2025b. Improving chain-of-thought reasoning via quasi-symbolic abstractions. In *Annual Meeting of the Association for Computational Linguistics*.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. 2024. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15504–15522, Bangkok, Thailand. Association for Computational Linguistics.

S Seals and Valerie Shalin. 2024. Evaluating the deductive competence of large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8614–8630, Mexico City, Mexico. Association for Computational Linguistics.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2022. Language models are multilingual chain-of-thought reasoners. *Preprint*, arXiv:2210.03057.

Samuel Soo, Chen Guang, Wesley Teng, Chandrasekaran Balaganesh, Tan Guoxian, and Yan Ming. 2025. Interpretable steering of large language models with feature guided activation additions. *Preprint*, arXiv:2501.09929.

Niklas Stoehr, Kevin Du, Vésteinn Snæbjarnarson, Robert West, Ryan Cotterell, and Aaron Schein. 2024. Activation scaling for steering and interpreting language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8189–8200, Miami, Florida, USA. Association for Computational Linguistics.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2024. Steering language models with activation engineering. *Preprint*, arXiv:2308.10248.

Marco Valentino, Geonhee Kim, Dhairya Dalal, Zhixue Zhao, and André Freitas. 2025. Mitigating content effects on reasoning in language models through fine-grained activation steering. *arXiv preprint arXiv:2505.12189*.

Wentao Wan, Zhuojie Yang, Yongcan Chen, Chenglin Luo, Ruilin Wang, Kehao Cai, Nan Kang, Liang Lin, and Keze Wang. 2025. Sr-fot: A syllogistic-reasoning framework of thought for large language models tackling knowledge-based reasoning tasks. *Preprint*, arXiv:2501.11599.

Yiming Wang, Zhuosheng Zhang, and Rui Wang. 2023. Meta-reasoning: Semantics-symbol deconstruction for large language models. *ArXiv*, abs/2306.17820.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

A Implementation Details

Software and Hardware All experiments were conducted using the Hugging Face Transformers library. Both training and inference were performed using half-precision (fp16) arithmetic on an NVIDIA RTX A6000 GPU.

Models In our experimental setting, as introduced in §3.1, we propose different models (detailed in Table 3). We choose the generation temperature for (mostly) deterministic outputs. The other parameters are left unchanged as recommended by the official resources.

Model	Version
Qwen-2.5-7B	Qwen/Qwen2.5-7B-Instruct
Qwen-3-14B	Qwen/Qwen3-14B
Gemma-2-9B	google/gemma-2-9b-it
Gemma-3-12B	google/gemma-3-12b-it
Mistral-7B	mistralai/Mistral-7B-Instruct-v0.3
Ministral-14B	mistralai/Ministral-3-14B-Instruct-2512

Table 3: List the versions of the models proposed in this work, which can be found on huggingface.co. We used all the default configurations proposed in the repositories for each model.

A.1 Steering Layer Selection

Optimal steering layers are identified empirically by analyzing the geometric separation between positive and negative target abstract representations. For each content-laden syllogism, we identify its positive target (an abstract syllogism with the same validity) and negative counterfactual (an abstract syllogism with opposite validity) using the matching procedure described in §2.3.

For each layer ℓ, we compute the average cosine similarity between these positive and negative target pairs:

$$s_\ell = \frac{1}{N} \sum_{i=1}^{N} \cos(a_\ell(y_i^+), a_\ell(y_i^-)) \qquad (14)$$

where N is the number of content-laden examples, y_i^+ is the positive abstract target for example i, and y_i^- is the corresponding negative counterfactual.

Layers with the lowest similarity s_ℓ exhibit the clearest separation between validity classes in the abstract reasoning space, making them optimal targets for steering. These layers are typically found in the middle layers (second or third quarter) of the network, where abstract concepts have formed but have not yet collapsed into task-specific outputs.

Model	Selected Layers
`Qwen-2.5-7B`	[18, 19, 20, 21, 22]
`Qwen-3-14B`	[24, 25, 26, 27, 28, 29]
`Gemma-2-9B`	[21, 22, 23, 24, 25]
`Gemma-3-12B`	[24, 25, 26, 27, 28, 29]
`Mistral-7B`	[13, 14, 15, 16, 17]
`Ministral-3-14B`	[20, 21, 22, 23, 24, 25]

Table 4: Optimal steering layers selected via similarity analysis (0-indexed).

Appendix J, Figure 15 visualizes this layer-wise analysis across all models.

A.2 Abstractor Architecture Details

Each Abstractor f_ℓ is implemented as a two-headed Multi-Layer Perceptron with the following architecture:

Shared Backbone A 3-layer feedforward network processes the input activation:

- Layer 1: Linear(d, 1024) + LayerNorm + LeakyReLU(0.01) + Dropout(0.1)
- Layer 2: Linear(1024, 1024) + LayerNorm + LeakyReLU(0.01) + Dropout(0.1)
- Layer 3: Linear(1024, 1024) + LayerNorm + LeakyReLU(0.01)

Direction Head Predicts the normalized direction vector:

- Linear(1024, 512) + ReLU
- Linear(512, d) + L2 Normalization

Magnitude Head Predicts the scalar magnitude:

- Linear(1024, 512) + ReLU
- Linear(512, 1) + Softplus

The final prediction combines both: $\hat{a}_\ell(x) = \hat{m}_\ell \cdot \hat{d}_\ell$.

Training Configuration Abstractors are trained using AdamW optimizer with learning rate 5×10^{-4} and weight decay 10^{-3}. We use batch size 128 with gradient clipping (max norm 1.0) and ReduceLROnPlateau scheduling (patience=10, factor=0.5). Training runs for up to 150 epochs with early stopping (patience=20). The contrastive margin threshold is set to 0.2, with loss weights $\lambda_{\text{repel}} = 0.75$ and $\lambda_{\text{mag}} = 1.0$.

A.3 SFT Implementation Details

We employ PiSSA (Principal Singular values and Singular vectors Adaptation) for parameter-efficient fine-tuning, using the following configuration:

LoRA Parameters

- Rank: $r = 16$
- Alpha: $\alpha = 16$
- Dropout: 0.05
- Target modules: query, key, value, and output projection layers
- Initialization: PiSSA (SVD-based initialization for faster convergence)

Training Data We train on the English training set, using both content-laden and abstract examples. For n content-abstract pairs in a fold, we create $2n$ training examples by including each variant separately. These examples are shuffled together during training, ensuring the model learns to handle both content-rich and symbolic inputs. This doubles the effective training set size compared to approaches that use only content-laden examples.

Training Hyperparameters Learning rate and epochs vary by model to account for differences in architecture and baseline performance:

- Qwen-2.5-7B: lr=10^{-5}, 2 epochs
- Gemma-2-9B: lr=10^{-5}, 1 epoch
- Mistral-7B: lr=10^{-5}, 5 epochs

All models use batch size 4 with gradient accumulation over 2 steps (effective batch size 8). We apply gradient clipping (max norm 1.0) and early stopping based on validation loss (patience=10 epochs). The training objective is standard cross-entropy loss on ground truth labels.

Label Masking To train the model to predict only the answer token, we mask all prompt tokens in the loss calculation. The system identifies where the answer begins by searching for "Valid" or "Invalid" tokens in the tokenized sequence, then sets all preceding positions to -100 (ignored index). If the answer position cannot be located, we conservatively mask the first 90% of the sequence.

A.4 CoT Implementation Details

Chain-of-Thought prompting encourages models to reason step-by-step before providing a final answer. We use the following prompt structure:

> *"Evaluate the logical validity of the following syllogism by reasoning step-by-step. First, analyze each premise carefully. Then, determine whether the conclusion logically follows from the premises. Finally, state your answer as either 'valid' or 'invalid'. Ignore the meaning, realism, or plausibility of the statements. Let's think step by step (be concise and conclude within a few sentences):"*

Generation Parameters We use greedy decoding (temperature=0) with max_new_tokens=512 to allow sufficient space for reasoning. The system prompt varies by model family but generally presents the assistant as skilled in logical reasoning.

Answer Extraction Since CoT responses contain both reasoning and the final answer, we extract the validity prediction using pattern matching. The system searches for explicit markers like "final answer", "therefore", or "the syllogism is" followed by "valid" or "invalid". If no clear marker exists, we examine the last 200 characters for standalone validity keywords.

Observed Failure Mode Despite step-by-step reasoning, CoT frequently inherits the same belief bias present in direct answers. Consider this example where the model was asked to evaluate an invalid syllogism:

> **Premises:** All cows are mammals. Some mammals are not birds.
> **Conclusion:** No birds are cows.
> **Ground Truth:** Invalid
>
> **Model Response:** *"The given syllogism is a classic example of a valid deductive argument. [...] Since cows are mammals and there are mammals that are not birds, it is impossible for a bird to be a cow because a bird is not a mammal. Therefore, the given syllogism is valid."*
>
> **Error:** The model incorrectly validates an invalid argument by confusing the logical structure. The premises do not support the conclusion: knowing that some mammals aren't birds tells us nothing about whether birds can be cows. The model's reasoning conflates semantic plausibility (birds obviously aren't cows in reality) with logical validity.

This failure pattern–where the model generates seemingly logical reasoning that ultimately justifies a semantically plausible but logically unsupported conclusion (or vice versa)–appears consistently across belief-conflict cases, suggesting that CoT prompting alone does not resolve the underlying content effect.

B Dataset and Translation Details

B.1 Language Selection Rationale

Our evaluation suite comprises ten languages selected to maximize typological and representational diversity across three key axes:

Family and Syntax The languages span four major families: Indo-European (English, French, Spanish, Italian, German, Russian, Bengali), Sino-Tibetan (Chinese), Dravidian (Telugu), and Niger-Congo (Swahili). This selection includes varying

syntactic orders–SVO (English, Chinese, Swahili) and SOV (Bengali, Telugu)–forcing the model to locate logical terms in different sequential positions. This tests whether the Abstractors target word order patterns or deeper structural abstractions.

Script and Tokenization We include languages that challenge the tokenizer differently: shared Latin subwords (Romance/Germanic), distinct scripts (Cyrillic for Russian, Bengali script, Telugu script), and logographic systems (Chinese). If steering relied on token-level patterns rather than semantic abstractions, we would expect dramatic performance drops when script systems change. The robust transfer to Chinese and Russian (§4.2) demonstrates that the intervention operates above the token level.

Resource Stratification High-resource languages (French, Spanish, Italian, German, Russian, Chinese) receive substantial exposure during pre-training, resulting in well-structured latent representations. Low-resource languages (Swahili, Telugu, Bengali) receive limited exposure, yielding noisier representations. This stratification allows us to disentangle steering mechanism quality from base model representation quality: if steering fails on LRLs, is it because the mechanism itself is weak, or because the underlying representations are insufficient? Our results (§4.2) support the latter interpretation.

B.2 Translation and Quality Check

We perform a three-stage quality control to preserve both *form* (logical structure, quantifiers, polarity) and *content* (lexical, named categories): **Structural and lexical guards** We execute template-level invariants on determiners/quantifiers (*all/no/some/some... not*), refuse contradiction, and check preservation for key nouns via bilingual lexicons. We require high semantic similarity between source and back-translation (SBERT cosine ≥ 0.90) and surface consistency (chrF ≥ 0.70). **Bilingual review** Native/near-native reviewers verify *(i)* faithful rendering of quantifiers, *(ii)* absence of idiomatic changes that could alter plausibility, and *(iii)* strict isomorphism of logical form. Disagreements trigger a corrective re-prompt with targeted constraints. **Invariance check** We validate that *validity* and *plausibility* labels remain unchanged after translation/back-translation.

Language	SBERT Cosine	chrF	Pass Rate (%)
English (en)	0.94	0.81	100.0
German (de)	0.92	0.76	97.8
Spanish (es)	0.93	0.77	98.2
French (fr)	0.91	0.74	96.5
Italian (it)	0.93	0.79	97.1
Russian (ru)	0.90	0.72	95.4
Chinese (zh)	0.89	0.71	94.8
Swahili (sw)	0.91	0.70	93.2
Bengali (bn)	0.90	0.73	94.6
Telugu (te)	0.88	0.69	92.5
Average	**0.91**	**0.74**	**95.8**

Table 5: Translation quality metrics and pass rates per language.

C Design Rationale and Class-Specific Transformations

C.1 Unified vs. Conditional Architecture

Preliminary analysis revealed that valid and invalid input syllogisms produce distinct activation patterns, suggesting that optimal transformations should be class-specific. While this observation motivates a conditional approach—using a classifier to gate separate Abstractors for each validity class—we opt for a **unified single-model architecture** in the main experiments for three reasons:

Error Propagation Conditional systems introduce cascading failure modes: classifier errors compound with transformation errors, creating a failure ceiling determined by the weaker component.

Feasibility Reliable validity probing, with performance matching the ones of the class-specific Abstractors, presupposes that the base model already encodes near-perfect internal representations distinguishing valid from invalid reasoning. However, if this was the case, it would be an important finding by itself. Our experiments confirm that validity probes achieve $\approx 95\%$ accuracy on English but degrade substantially in multilingual settings, limiting the practical effectiveness of conditional approaches.

Efficiency Auxiliary classifiers add computational overhead and increase system complexity. Rather than relying on explicit gating, our unified Abstractor uses contrastive loss to *implicitly* route activations based on their input geometry, automatically mapping valid and invalid inputs to their respective abstract manifolds.

C.2 Oracle Validation: Class-Specific Upper Bound

To validate that our approach's limitations stem from routing rather than transformation quality, we implemented class-specific Abstractors. When applied in an "oracle" setting–where we use ground truth labels to select the corresponding valid or invalid Abstractor–we achieved $\sim 100\%$ **accuracy across HRLs for all models**. This confirms that the observed failure modes arise solely from the challenge of separating valid and invalid logic paths.

To quantify the theoretical upper-bound without an oracle, we trained validity probes on the last token activations of each steering layer. We implemented this using a Support Vector Machine (SVM) with an RBF kernel, applied to standardized activation vectors with a stratified 80/20 train-test split.

Since the class-specific Abstractors gain perfect accuracy when knowing the ground truth, the bottleneck is strictly the probe. Therefore, the results shown in Figure 6 represent the *hard upper bound* for a possible class-specific implementation. As observed, accuracy on non-english languages, in particular on LRLs, is only acceptable when the probe is trained with few-shot examples in the target languages. The English-trained probe exhibits a mild degradation on HRLs and a severe drop on LRLs, undermining the zero-shot generalisation objective of our design.

This analysis confirms our architectural choice: while class-specific transformations represent a theoretical upper bound, the unified approach is better suited for zero-shot generalisation, offering the best practical trade-off between performance, robustness, and system complexity.

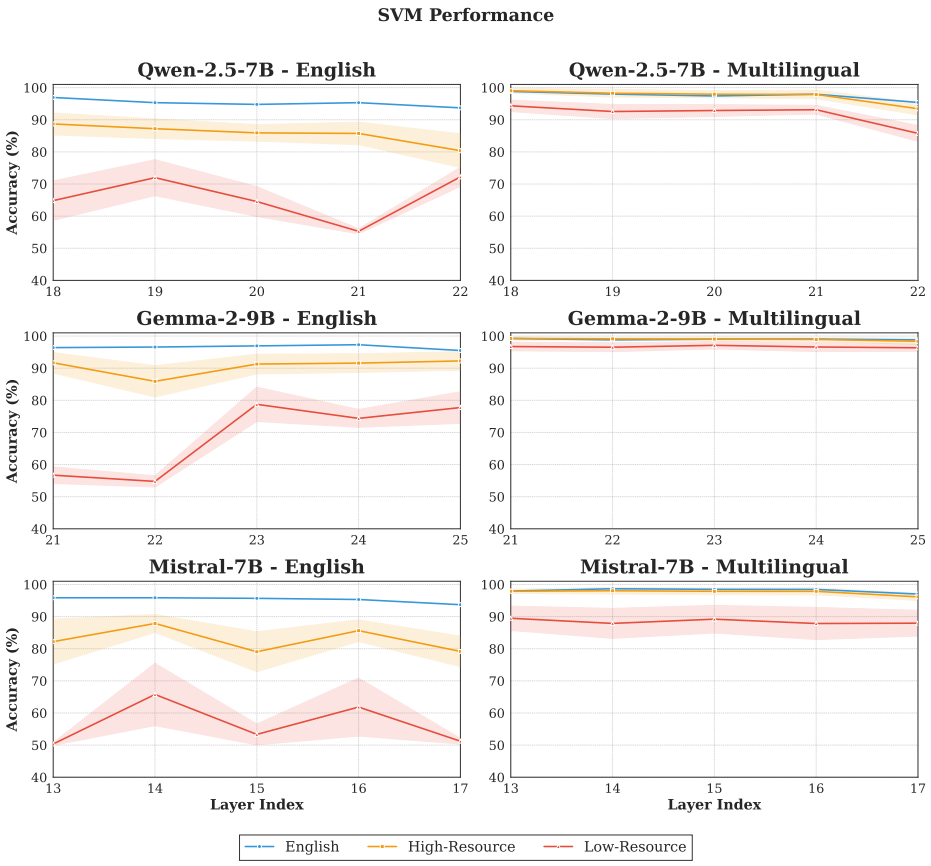


Figure 6: Validity probe (SVM with RBF kernel) accuracy across selected steering layers for `Qwen-2.5-7B`, `Gemma-2-9B` and `Mistral-7B`. The probe trained only on English maintains acceptable accuracy on HRLs but degrades severely on LRLs. Few-shot probes achieve optimal cross-lingual performance but undermine the zero-shot objective of our framework.

D Detailed Baseline Abstract Results

Model	Categories				Global	
	VP	VI	IP	II	Acc	BPA
`Qwen-2.5-7B`	71.59	71.77	94.94	94.84	83.27	83.15
`Qwen-3-14B`	80.06	81.26	99.28	98.28	89.71	88.72
`Gemma-2-9B`	85.07	85.45	82.33	83.14	83.99	83.81
`Gemma-3-12B`	98.71	98.70	88.57	89.99	93.99	93.32
`Mistral-7B`	100.00	100.00	70.91	70.25	85.29	85.01
`Ministral-14B`	82.92	82.85	87.98	86.55	85.07	84.49

Table 6: Performance on abstract subset. VP/II are belief-consistent; VI/IP are belief-conflict. These results represent the upper bound that steering aims to approximate when processing content-laden inputs. The convergence of Accuracy (Acc) and BPA demonstrates minor content effect (content is removed), with minor variations due to statistical noise.

E Detailed Language-Specific Results

E.1 Qwen-2.5-7B

Language	Baseline						Steered					
	Categories				Global		Categories				Global	
	VP	VI	IP	II	Acc	BPA	VP	VI	IP	II	Acc	BPA
English (en)	94.5	31.1	90.6	98.0	78.6	50.8	99.4	98.8	93.2	91.7	95.8	95.4
French (fr)	87.9	21.7	80.8	93.5	71.0	43.0	90.7	90.6	91.3	91.6	91.0	90.8
Spanish (es)	79.2	16.2	85.0	97.6	69.6	43.3	80.6	82.8	93.3	95.3	88.0	87.9
Italian (it)	80.0	14.6	82.9	96.3	68.5	41.5	84.9	86.4	88.0	91.9	87.8	86.8
German (de)	67.5	15.8	90.2	95.9	67.4	48.0	82.1	83.9	89.7	92.0	86.9	86.8
Russian (ru)	70.8	10.0	82.5	97.2	65.2	40.6	72.2	71.9	86.3	90.9	80.4	78.4
Chinese (zh)	86.2	15.8	83.3	97.2	70.7	40.9	81.2	78.3	94.6	93.2	86.8	86.2
Bengali (bn)	72.1	27.9	79.9	87.8	67.0	49.5	62.9	59.3	82.2	88.3	73.2	69.7
Swahili (sw)	37.1	23.8	86.8	91.5	59.8	54.4	38.5	29.7	85.8	88.2	60.6	57.2
Telugu (te)	70.0	54.6	73.1	79.7	69.4	61.7	59.9	55.7	80.8	84.4	70.2	67.5

Table 7: Detailed accuracy breakdown in `Qwen-2.5-7B` at best α (highest BPA) per language. Left: Baseline model performance. Right: Performance with steering. VP/II are belief-consistent; VI/IP are belief-conflict.

E.2 Qwen-3-14B

Language	Baseline						Steered					
	Categories				Global		Categories				Global	
	VP	VI	IP	II	Acc	BPA	VP	VI	IP	II	Acc	BPA
English (en)	83.1	59.7	94.1	98.0	83.7	72.3	99.7	99.1	96.8	96.0	97.9	97.8
French (fr)	89.2	51.7	88.0	97.2	81.6	62.5	96.9	92.2	92.6	92.7	93.6	91.4
Spanish (es)	81.7	42.9	87.6	97.2	77.4	58.7	97.5	90.3	92.9	95.8	94.1	89.4
Italian (it)	84.2	42.9	88.5	97.2	78.2	58.7	96.7	87.9	93.3	95.5	93.4	88.2
German (de)	87.9	46.7	84.2	95.1	78.5	58.0	93.3	84.6	93.7	93.9	91.4	87.3
Russian (ru)	79.6	31.7	84.6	97.6	73.4	51.1	92.4	73.1	89.5	93.6	87.2	76.9
Chinese (zh)	80.8	37.9	87.6	98.0	76.2	55.9	96.0	88.9	91.3	94.3	92.6	88.0
Bengali (bn)	64.6	25.4	91.9	97.2	69.8	54.3	88.6	75.1	87.5	88.9	85.0	78.7
Swahili (sw)	15.4	11.7	94.9	95.1	54.3	53.2	27.9	24.4	89.6	90.9	58.2	56.8
Telugu (te)	56.2	25.0	90.6	97.2	67.3	54.6	86.4	74.7	84.5	88.8	83.6	76.9

Table 8: Detailed accuracy breakdown in `Qwen-3-14B` at best α (highest BPA) per language. Left: Baseline model performance. Right: Performance with steering. VP/II are belief-consistent; VI/IP are belief-conflict.

E.3 Gemma-2-9B

Language	Baseline						Steered					
	Categories				Global		Categories				Global	
	VP	VI	IP	II	Acc	BPA	VP	VI	IP	II	Acc	BPA
English (en)	92.1	45.2	75.4	93.1	76.5	51.8	99.7	98.7	97.5	96.8	98.2	98.0
French (fr)	90.4	45.0	70.1	90.7	74.2	49.7	98.1	96.5	90.6	95.8	95.3	92.1
Spanish (es)	91.7	40.4	70.1	91.9	73.7	46.8	97.8	96.4	93.0	95.7	95.7	93.8
Italian (it)	90.4	35.8	67.5	90.2	71.2	43.6	98.3	91.4	93.6	95.4	94.7	90.5
German (de)	87.1	30.4	67.1	90.2	68.8	41.4	96.1	93.9	92.2	94.4	94.2	92.0
Russian (ru)	83.8	30.0	68.4	93.5	69.1	41.8	90.7	82.6	87.0	90.8	87.8	82.6
Chinese (zh)	85.4	24.2	71.4	93.5	68.8	40.1	92.9	81.1	90.3	93.8	89.5	82.7
Bengali (bn)	70.8	36.2	74.8	92.3	68.7	50.8	73.3	72.6	91.7	92.3	82.5	82.0
Swahili (sw)	57.1	22.9	69.7	87.0	59.3	44.0	54.7	48.9	79.1	82.9	66.4	63.2
Telugu (te)	61.7	21.2	79.5	96.8	64.9	46.2	60.1	58.1	93.7	95.1	76.8	75.4

Table 9: Detailed accuracy breakdown in `Gemma-2-9B` at best α (highest BPA) per language. Left: Baseline model performance. Right: Performance with steering. VP/II are belief-consistent; VI/IP are belief-conflict.

E.4 Gemma-3-12B

Language	Baseline						Steered					
	Categories				Global		Categories				Global	
	VP	VI	IP	II	Acc	BPA	VP	VI	IP	II	Acc	BPA
English (en)	99.7	87.5	79.7	89.0	89.0	79.4	98.9	97.3	97.7	98.3	98.0	97.0
French (fr)	98.8	85.4	65.0	87.4	84.3	69.2	95.3	91.8	94.4	95.3	94.2	92.2
Spanish (es)	98.3	85.4	59.0	87.8	82.8	65.5	92.1	92.9	95.2	96.8	94.2	93.9
Italian (it)	97.5	84.6	62.8	89.8	83.8	67.1	93.1	92.8	95.0	97.3	94.5	93.3
German (de)	94.6	78.8	65.0	86.2	81.2	66.2	94.2	91.5	93.5	95.5	93.7	91.5
Russian (ru)	96.7	70.0	56.4	88.6	78.1	55.1	91.9	83.2	90.9	94.3	90.1	84.6
Chinese (zh)	96.7	72.5	59.4	89.4	79.7	58.1	95.3	86.0	90.3	94.8	91.6	85.3
Bengali (bn)	97.5	82.9	56.0	86.2	80.8	62.7	89.2	87.8	88.5	93.1	89.7	87.0
Swahili (sw)	74.2	46.7	55.1	80.5	64.3	47.3	59.6	49.3	83.9	88.1	70.2	65.2
Telugu (te)	95.0	71.7	68.4	86.2	80.4	63.9	92.1	87.4	85.3	88.5	88.3	84.8

Table 10: Detailed accuracy breakdown in`Gemma-3-12B` at best α (highest BPA) per language. Left: Baseline model performance. Right: Performance with steering. VP/II are belief-consistent; VI/IP are belief-conflict.

E.5 Mistral-7B

Language	Baseline						Steered					
	Categories				Global		Categories				Global	
	VP	VI	IP	II	Acc	BPA	VP	VI	IP	II	Acc	BPA
English (en)	99.7	41.1	35.7	84.8	65.4	30.2	99.6	96.8	96.5	96.1	97.2	96.1
French (fr)	96.7	31.4	45.3	84.2	64.7	31.0	94.2	87.4	82.3	87.3	87.8	82.6
Spanish (es)	93.8	25.3	45.5	81.3	61.8	29.6	89.0	84.4	82.8	82.9	84.8	82.8
Italian (it)	95.8	32.1	44.4	80.5	63.5	31.8	89.8	78.8	73.3	75.1	79.3	74.2
German (de)	94.6	27.5	38.9	78.4	60.1	28.1	89.3	85.5	79.2	83.5	84.4	81.0
Russian (ru)	93.8	26.6	34.6	79.2	58.9	26.0	79.8	66.7	61.0	68.2	69.0	62.0
Chinese (zh)	83.8	25.3	50.4	80.9	60.4	33.5	78.5	72.9	69.9	77.2	74.7	69.9
Bengali (bn)	71.2	53.6	54.9	64.7	61.1	52.7	72.9	63.6	61.2	69.5	66.8	61.0
Swahili (sw)	37.2	22.7	60.8	71.0	47.9	41.9	37.2	40.3	69.6	68.0	54.0	52.7
Telugu (te)	33.0	29.2	67.5	72.2	50.2	48.1	34.5	35.1	54.9	58.3	45.9	45.2

Table 11: Detailed accuracy breakdown in `Mistral-7B` at best α (highest BPA) per language. Left: Baseline model performance. Right: Performance with steering. VP/II are belief-consistent; VI/IP are belief-conflict.

E.6 Ministral-14B

Language	Baseline						Steered					
	Categories				Global		Categories				Global	
	VP	VI	IP	II	Acc	BPA	VP	VI	IP	II	Acc	BPA
English (en)	88.4	61.4	78.7	92.3	80.2	63.9	98.8	98.0	97.2	96.6	97.7	97.6
French (fr)	85.4	48.8	70.1	91.5	74.1	52.6	94.2	93.3	90.6	96.6	93.7	90.5
Spanish (es)	78.3	38.3	75.2	94.7	71.8	50.4	91.5	89.2	92.3	96.3	92.4	89.4
Italian (it)	79.2	41.2	70.1	90.2	70.3	49.9	93.6	91.5	90.9	92.7	92.2	90.4
German (de)	72.1	36.2	77.3	92.7	69.7	51.9	94.0	89.9	93.0	95.5	93.1	90.0
Russian (ru)	70.4	30.0	76.5	95.1	68.1	48.0	86.1	73.2	87.3	92.3	84.8	77.2
Chinese (zh)	72.1	25.4	70.9	95.9	66.2	42.5	85.6	71.7	93.3	96.6	86.8	79.3
Bengali (bn)	70.8	27.9	80.8	94.7	68.7	49.1	78.1	64.4	89.6	95.5	81.9	73.9
Swahili (sw)	29.2	15.8	85.5	89.0	54.9	50.3	29.7	23.9	95.9	95.4	61.2	59.6
Telugu (te)	71.2	38.3	73.9	90.2	68.5	51.7	72.1	57.4	95.3	95.7	80.1	74.1

Table 12: Detailed accuracy breakdown in `Ministral-14B` at best α (highest BPA) per language. Left: Baseline model performance. Right: Performance with steering. VP/II are belief-consistent; VI/IP are belief-conflict.

F Steering Strength Ablation Study

F.1 Qwen-2.5-7B

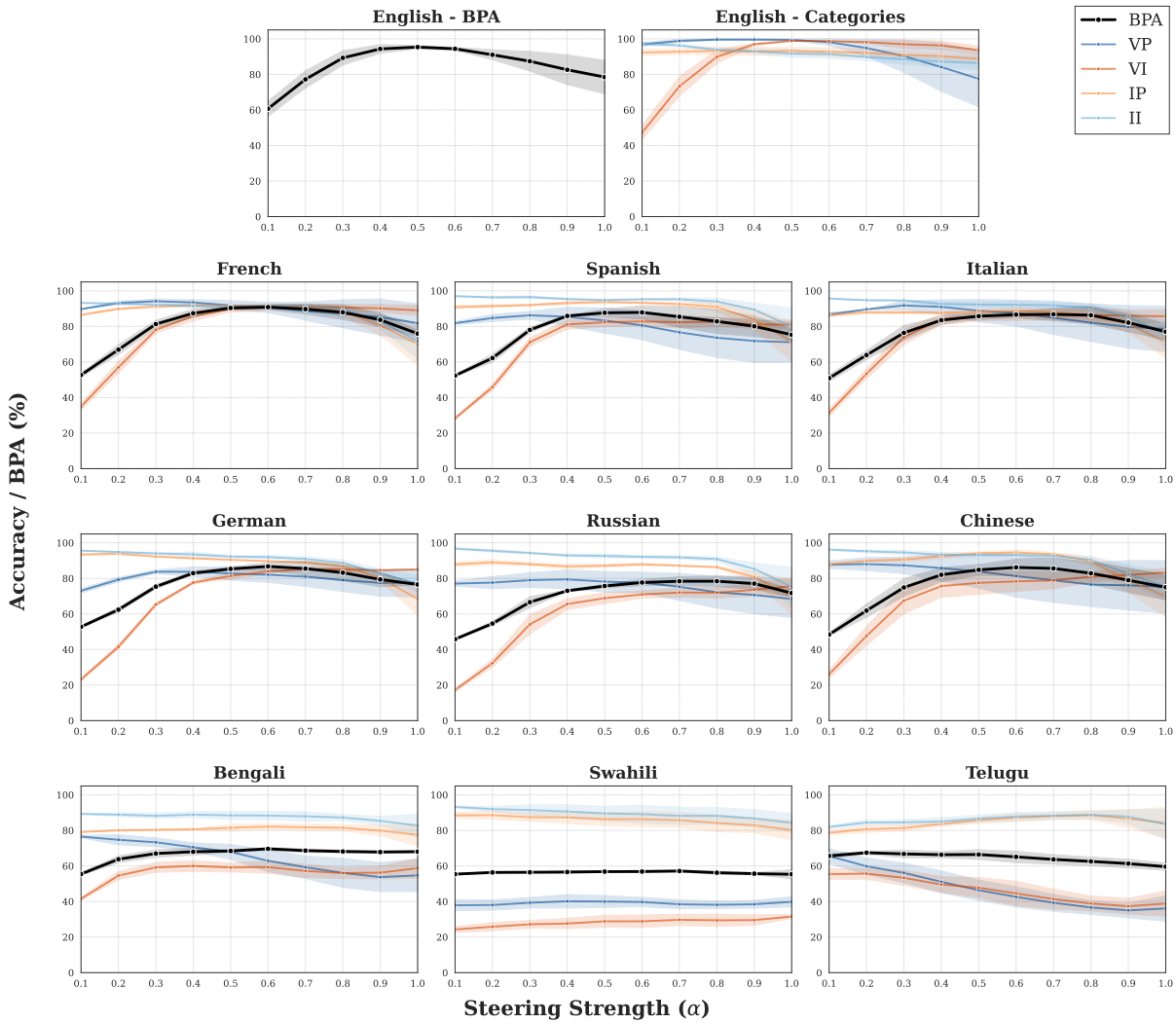


Figure 7: Steering strength (α) ablation study for `Qwen-2.5-7B`, showing per-category accuracy and BPA (black) across all languages. VP/II (blue) are belief-consistent; VI/IP (orange) are belief-conflict.

F.2 Qwen-3-14B

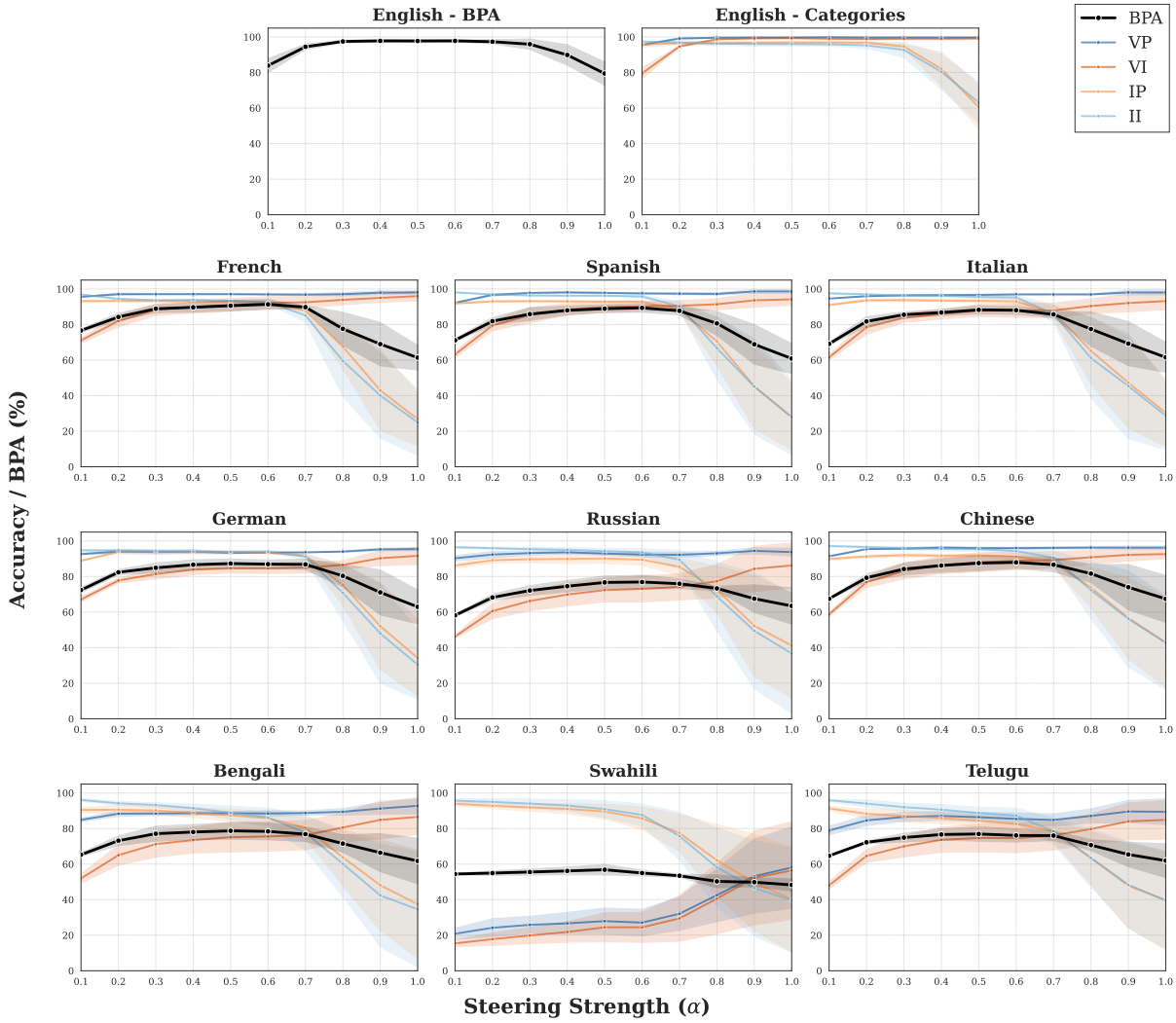


Figure 8: Steering strength (α) ablation study for `Qwen-3-14B`, showing per-category accuracy and BPA (black) across all languages. VP/II (blue) are belief-consistent; VI/IP (orange) are belief-conflict.

F.3 Gemma-2-9B

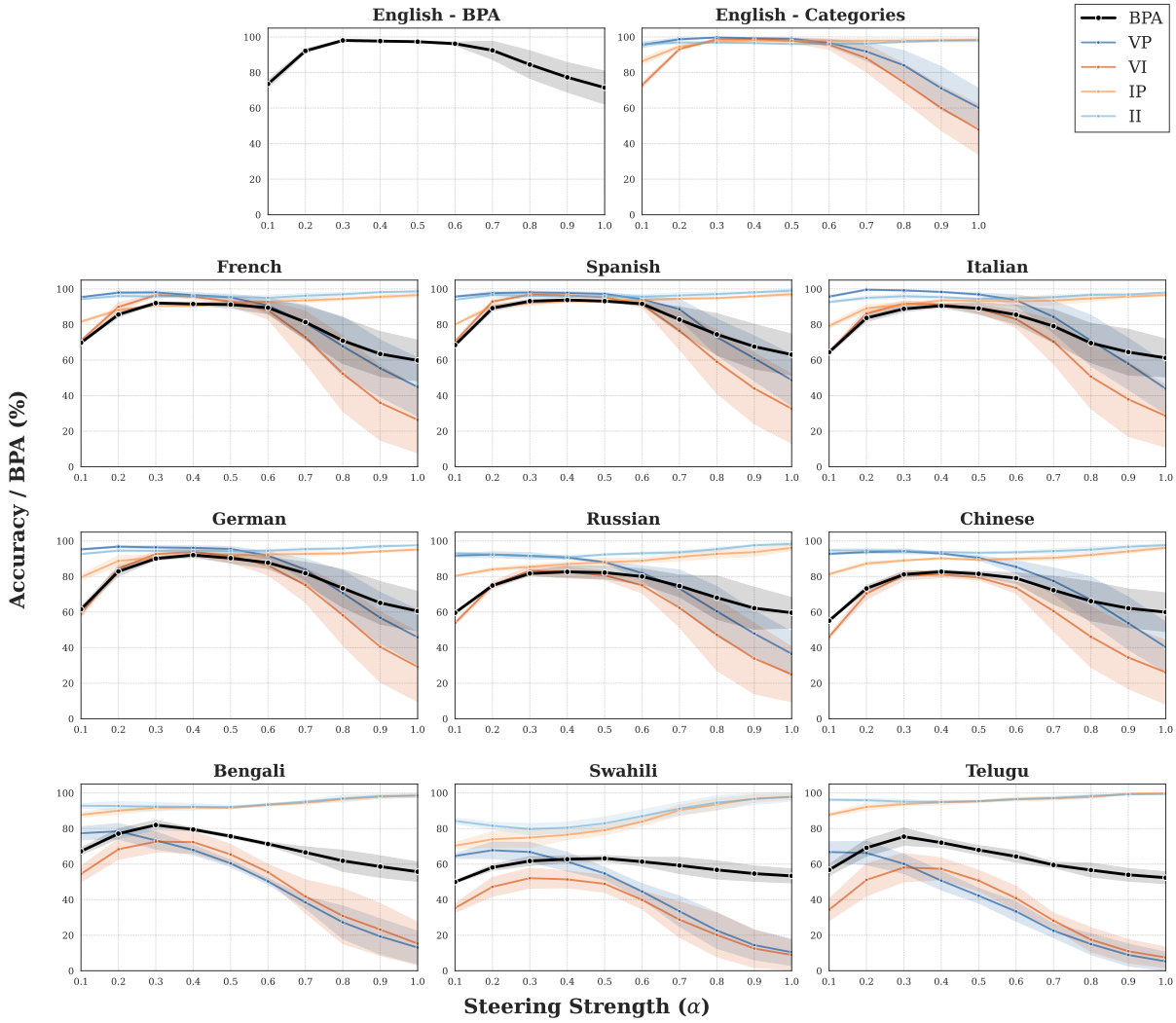


Figure 9: Steering strength (α) ablation study for `Gemma-2-9B`, showing per-category accuracy and BPA (black) across all languages. VP/II (blue) are belief-consistent; VI/IP (orange) are belief-conflict.

F.4 Gemma-3-12B

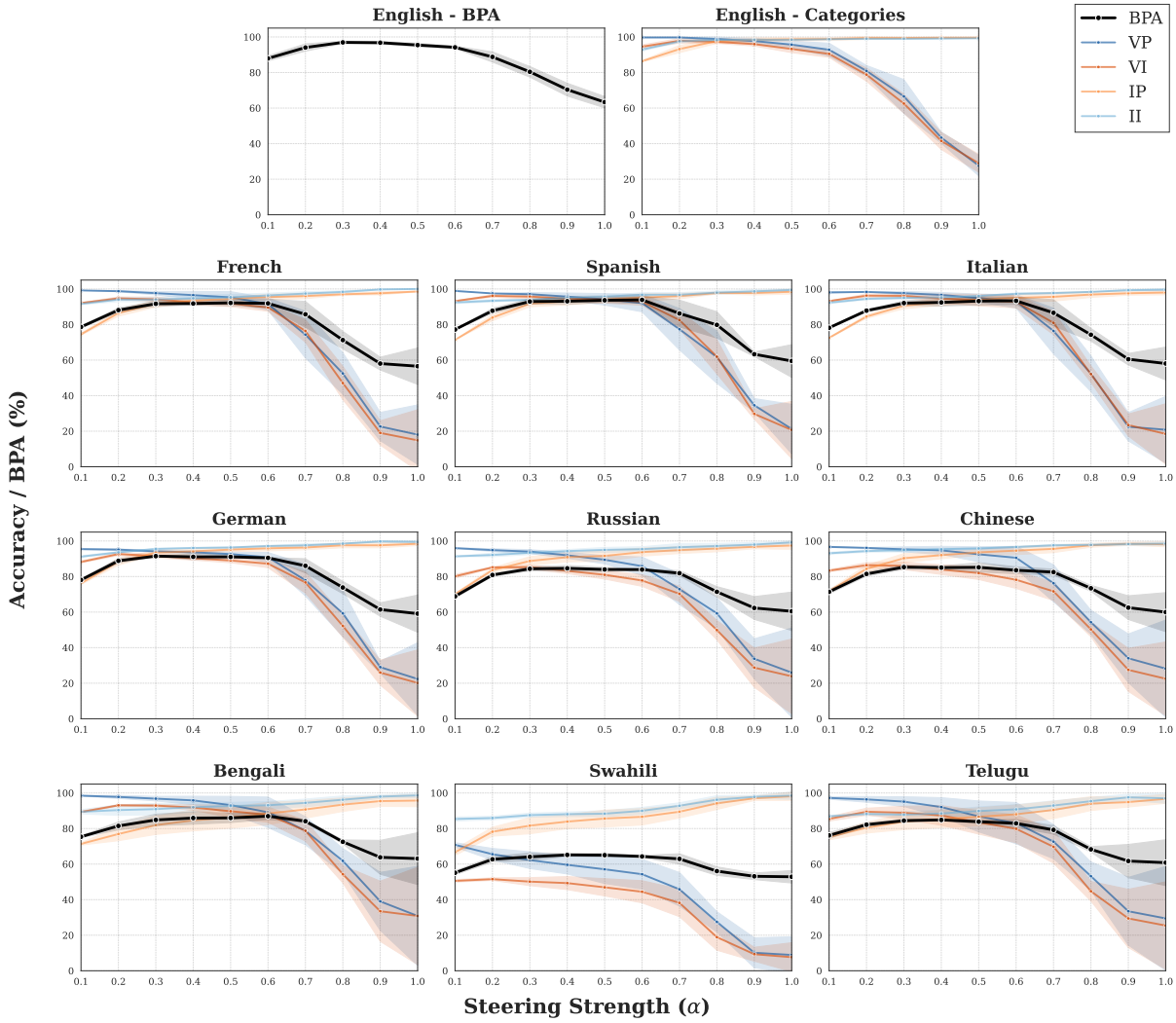


Figure 10: Steering strength (α) ablation study for `Gemma-3-12B`, showing per-category accuracy and BPA (black) across all languages. VP/II (blue) are belief-consistent; VI/IP (orange) are belief-conflict.

F.5 Mistral-7B

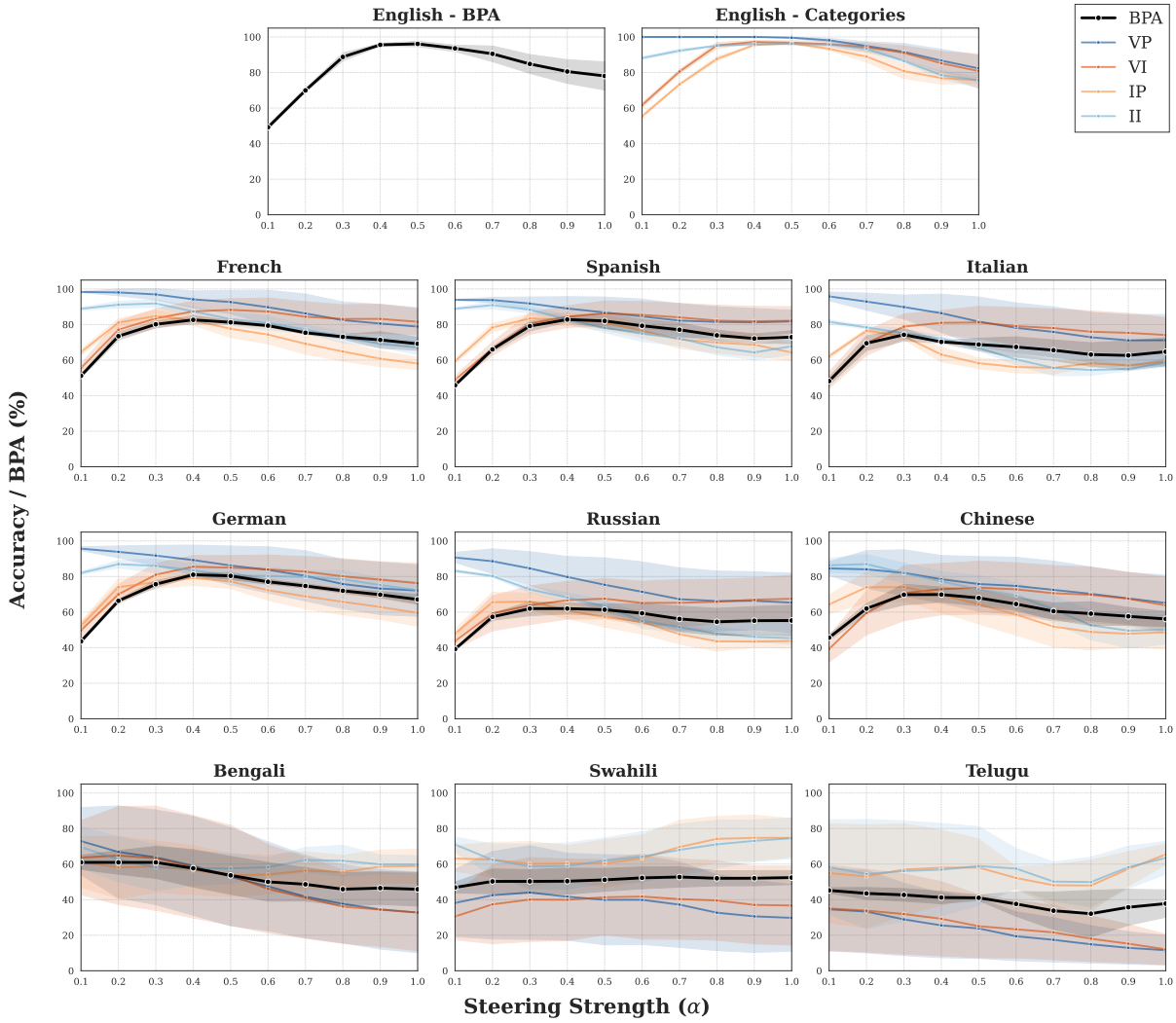


Figure 11: Steering strength (α) ablation study for `Mistral-7B`, showing per-category accuracy and BPA (black) across all languages. VP/II (blue) are belief-consistent; VI/IP (orange) are belief-conflict.

F.6 Ministral-14B

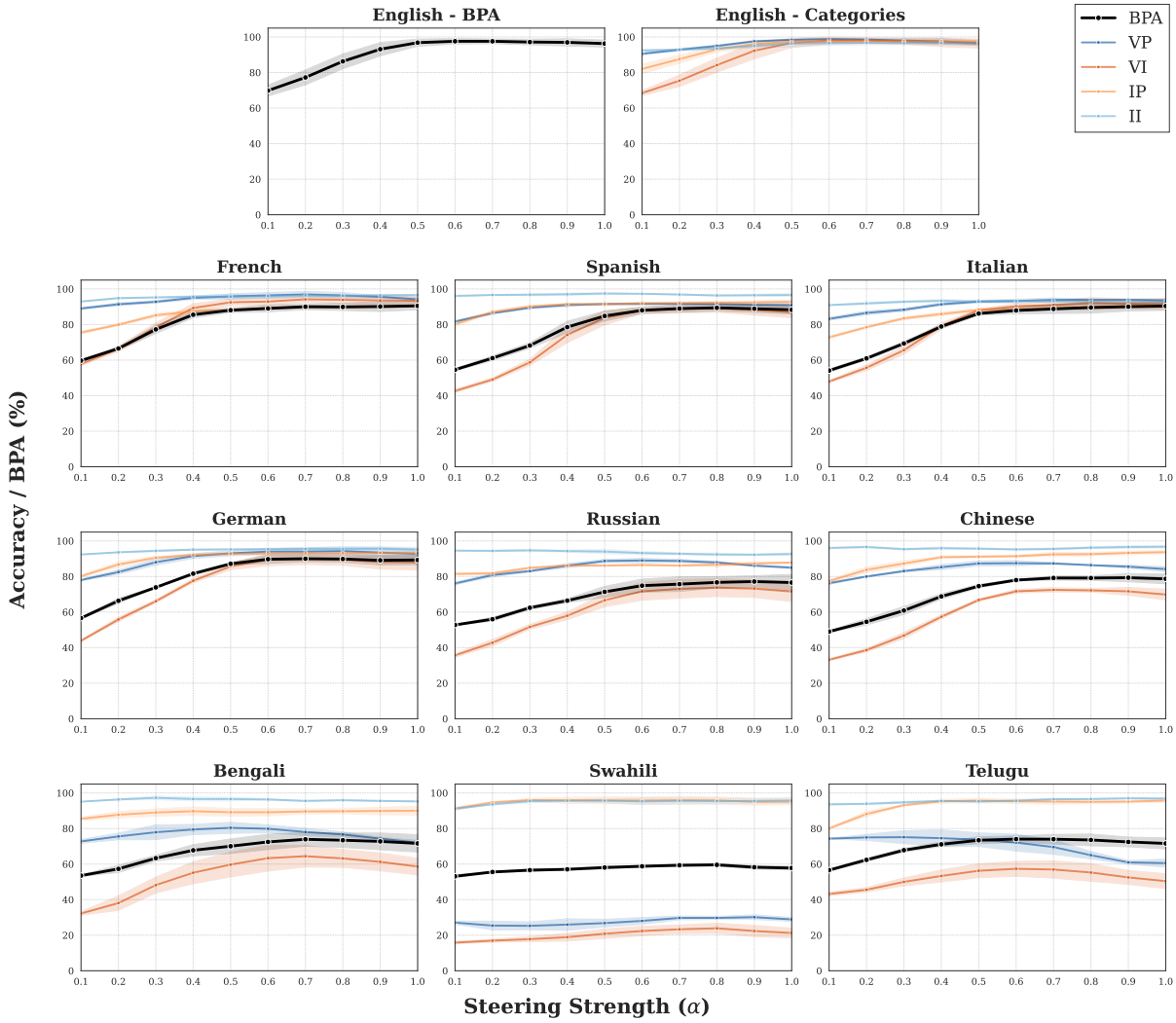


Figure 12: Steering strength (α) ablation study for `Ministral-14B`, showing per-category accuracy and BPA (black) across all languages. VP/II (blue) are belief-consistent; VI/IP (orange) are belief-conflict.

G Detailed Steering vs. SFT vs. CoT Results

Model	Method	English	HRLs	LRLs
Qwen-2.5-7B	Steering	94.99	85.43	65.30
	SFT	98.57 (+3.77%)	84.63 (-0.94%)	70.12 (+7.38%)
	CoT	81.65 (-14.04%)	57.73 (-32.43%)	55.00 (-15.77%)
Gemma-2-9B	Steering	98.25	89.65	76.62
	SFT	98.78 (+0.54%)	93.39 (+4.17%)	76.96 (+0.44%)
	CoT	69.92 (-28.83%)	58.54 (-34.71%)	53.58 (-30.08%)
Mistral-7B	Steering	94.77	77.71	56.49
	SFT	88.09 (-7.05%)	78.17 (+0.59%)	53.68 (-4.98%)
	CoT	34.25 (-63.86%)	34.20 (-55.99%)	49.97 (-11.55%)

Table 13: Comparison of Bias-Penalized Accuracy (BPA) across Activation Steering, Supervised Fine-Tuning (SFT), and Chain-of-Thought (CoT) prompting. Percentage changes relative to steering are shown in green (improvement) or red (degradation).

H Detailed PPL Results

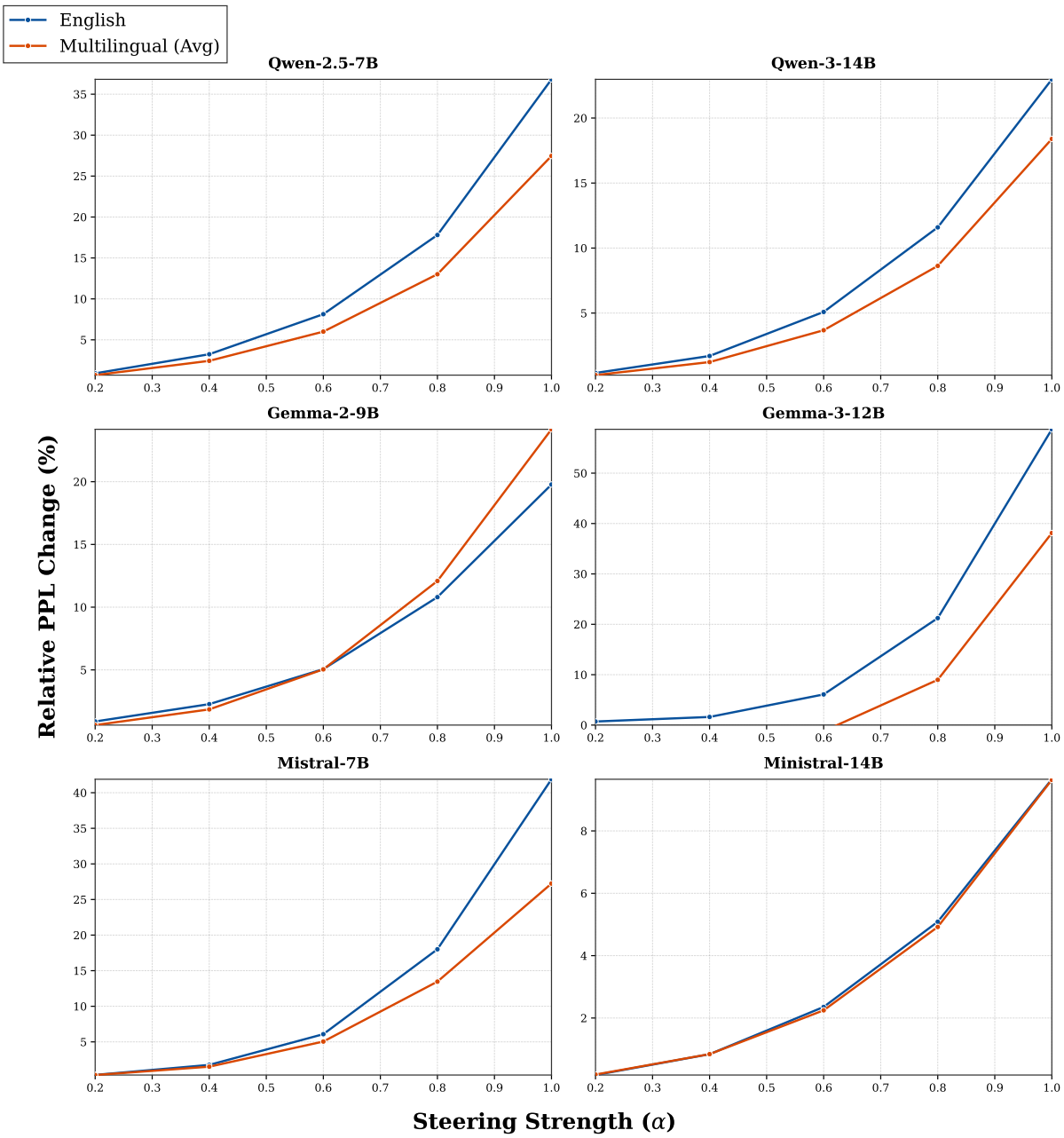


Figure 13: Perplexity degradation as a function of steering strength across all models and languages. At optimal steering configurations ($\alpha \approx 0.4 - 0.6$), fluency remains largely preserved. Significant degradation emerges only at aggressive steering levels ($\alpha \geq 0.8$).

I Detailed OOD Task Results

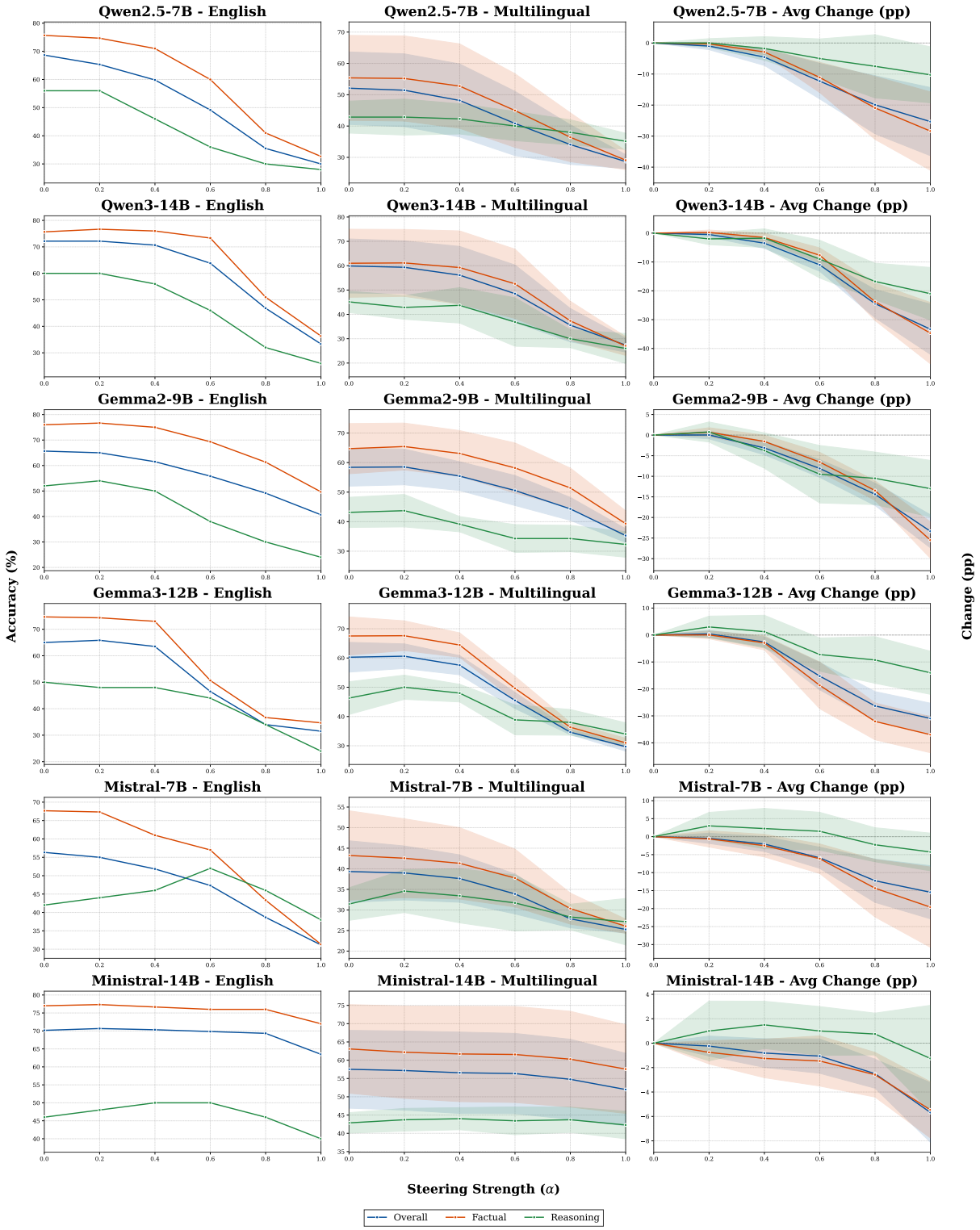
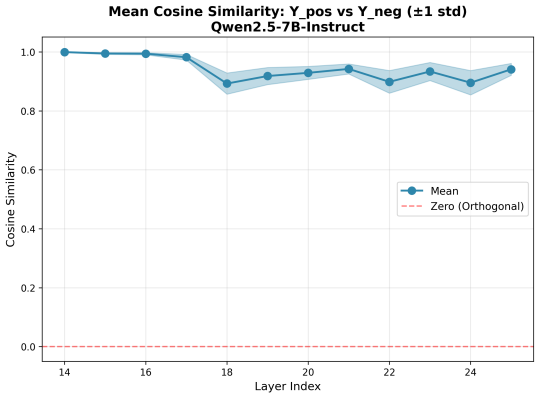
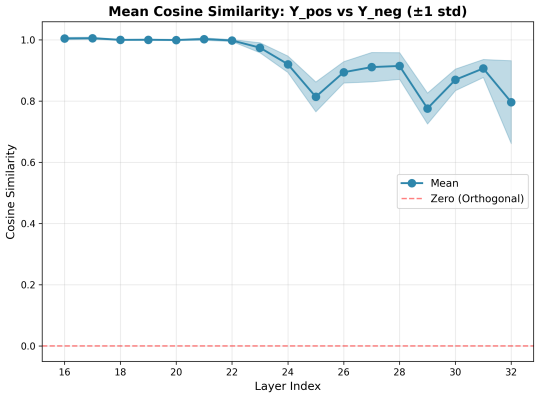


Figure 14: Performance impact on MMMLU subtasks across models. Factual knowledge tasks (orange) degrade sharply as steering suppresses semantic content, while Formal Logic tasks (green) often show more resilience. This divergence confirms the specificity of the intervention mechanism.

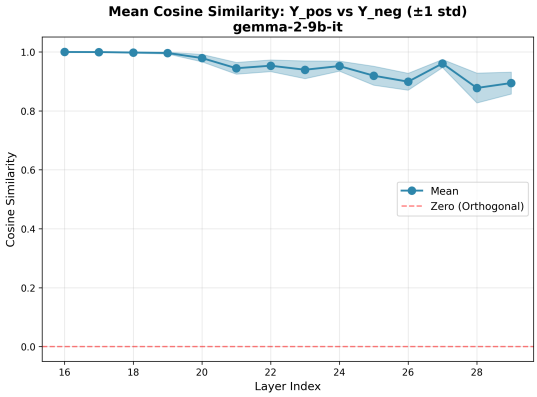
J Positive-Negative Cosine Similarity Analysis



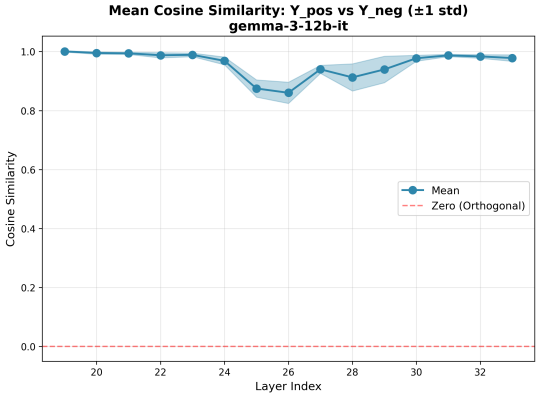
(a) Qwen2.5-7B



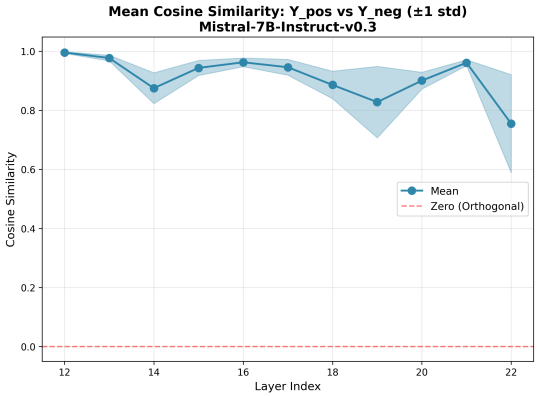
(b) Qwen3-14B



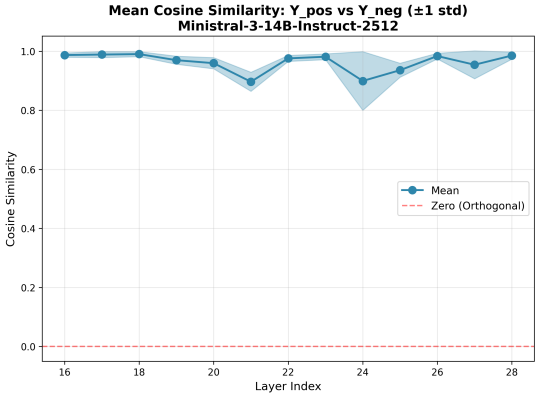
(c) Gemma-2-9B



(d) Gemma-3-12B



(e) Mistral-7B



(f) Ministral-3-14B

Figure 15: Layer-wise cosine similarity between positive and negative training target pairs across all models.