# Understanding Code-Switching in Multilingual LLMs through Mechanistic Interpretability

**Anonymous Authors**

## Abstract

This work investigates how different transformer models internally process French-English and Hindi-English code-switched (CS) input using mechanistic interpretability techniques. We analyse neuron activations and layer-wise representations across aligned monolingual and code-switched conditions. Our results indicate that code-switched representations are governed by the matrix language, which provides the grammatical structure, rather than forming a distinct computational category. Language-sensitive neuron responses are tightly localised at switch boundaries. Hindi-English switching consistently produces stronger matrix dominance and more localised switch-point responses than French-English switching, which we attribute to script discontinuity, linguistic distance, and differences in pre-training data composition. These findings contribute to a mechanistic understanding of how multilingual LLMs handle mixed-language input.

## 1 Introduction

Large language models (LLMs) have become increasingly capable in multilingual settings, and have demonstrated the ability to produce code-switched output. Despite this capability, the internal mechanisms underlying such behaviour remain poorly understood. This gap matters for both practical and scientific reasons, since whether a model should produce code-switched output ultimately depends on user preferences, and a clearer understanding of how code-switching is represented internally could inform more controllable and better-aligned multilingual systems.

Code-switching, defined as the alternation between languages within a sentence, or text by a multilingual speaker (Poplack, 1980), is common in bilingual communication. However, for a language model processing such input, this presents a non-trivial representational challenge: simultaneously managing multiple linguistic systems and their transitions. While prior work has extensively evaluated multilingual performance and generation quality, little attention has been paid to how LLMs internally process code-switched inputs at the level of individual components and representations.

In this work, this gap is addressed using mechanistic interpretability. We analyse how specific model components neurons, layers, and residual stream representations respond during language switches, and examine how internal representations evolve across network depth when processing mixed-language inputs.

Our main contributions are as follows [1]:

- Two parallel code-switching datasets for French-English and Hindi-English, each providing three aligned representations of the same content: code-switched, English-translated, and monolingual target-language versions.
- We present a neuron-level mechanistic analysis targeting code-switched input across multiple language pairs and model architectures.
- We demonstrate that code-switched representations are dominated by the matrix language rather than constituting a distinct representational category, and that this effect varies systematically with linguistic distance and model architecture.

## 2 Related Work

LLMs have demonstrated strong multilingual proficiency through shared cross-lingual representations, enabling transfer across languages and a wide range of multilingual tasks (Xu et al., 2025).

One such capability is the production of code-switched text – a phenomenon central to fluent

[1] Code and datasets for this work can be found at https://github.com/anonymous-arr-2026/arr-submission

multilingual communication. Because multilingual models are trained on diverse data that may include language mixing, they can intentionally alternate between languages when appropriately prompted, demonstrating the capability for controlled or intentional code-switching (Yong et al., 2023).

Despite this capability, even state-of-the-art multilingual LLMs frequently struggle with processing and generating code-switched text (Zhang et al., 2023), particularly when the languages involved are unevenly represented in training data (Yong et al., 2023). Generally, foreign tokens within otherwise monolingual inputs tend to degrade performance across language understanding and reasoning tasks (Mohamed et al., 2025), and code-switched prompts can produce poor quality outputs (Oh et al., 2025). Yet these findings provide little insight into how models internally process mixed-language inputs.

Mechanistic interpretability addresses this by investigating how internal components of transformer models contribute to specific behaviours. Recent work shows that linguistic information – syntax, semantics, and language identity – is reflected in layer-wise activation patterns, with semantically equivalent sentences across languages producing similar neuron activations (Zeng et al., 2025). In the same study, tasks such as language identification are linked to specific neurons or attention heads, suggesting that some functions can be traced to individual components within a model.

Mechanistic interpretability has recently been applied to study other multilingual behaviour in LLMs. For language confusion (unintended language mixing), mechanistic tools such as TunedLens and neuron attribution reveal that such errors arise primarily in the final transformer layers, and that modifying a small subset of neurons can reduce them (Nie et al., 2025). Similarly, a small set of language-specific neurons in early and late layers has been shown to govern language output, such that suppressing them redirects generated output toward English (Tang et al., 2024).

However, mechanistic interpretability studies remain limited in scope, and it has not yet been applied to intentional code-switching. Research in the code-switching landscape has focused on outputs and benchmarks rather than internal representations. Our work fills this gap by applying mechanistic techniques to analyse how LLMs process code-switched text and whether specific internal components can be linked to code-switching behaviour.

## 3 Methodology

### 3.1 Dataset Construction

We synthetically construct two datasets for Hindi–English (Hi-En) and French–English (Fr-En) code-switching, each containing 1,000 samples. Every sample is represented in three forms: a code-switched version, an English translation, and a native-language version. This choice of language pairs allows us to examine both cross-script (Devanagari–Latin) and same-script (Latin–Latin) code-switching across different multilingual settings.

Each dataset is evenly distributed across **four domains**: *Entertainment*, *Science*, *Sports*, and *News* (250 samples per domain), reflecting the diversity of contexts in which code-switching naturally occurs and reducing the risk of domain-specific bias.

**French–English Dataset.** The primary source is **SwitchLingua**, used for the *Entertainment*, *Science*, and *Sports* domains. It provides both *inter-* and *intra-sentential* code-switched text (Table 1) generated synthetically via a multi-agent LLM framework (Xie et al., 2025). For the *News* domain, we use the **French News** dataset (Kaggle[2]), a naturally occurring parallel corpus of French financial news summaries with English translations (published 2018–2021).

The construction pipeline loads data from each corpus, samples 250 instances per domain (capped at four sentences per sample to control size while preserving sentence length variation), and assembles parallel code-switched, English, and native-language versions. Code-switched samples are translated using **MarianMT** (Junczys-Dowmunt et al., 2018). For the French News data, parallel French-English text is available, and code-switching is generated by replacing selected French words (nouns, verbs, adjectives) with their English equivalents.

**Hindi–English Dataset.** The Hindi–English dataset is derived directly from the Fr–En dataset, reusing the same sentences and English words

[2] https://www.kaggle.com/datasets/arcticgiant/french-financial-news

where possible so that cross-lingual and model comparisons reflect language-pair differences rather than content variation.

First, pure English sentences are translated into Hindi using *ai4bharat/indictrans2-en-indic-1B*, with long sentences split at boundaries, translated independently, and concatenated. Second, code-switched sentences are generated by using *lingua* to identify English fragments in the Fr–En code-switched sentences, cross-checking these against the pure English source to remove misclassifications, and then prompting *google/gemma-3-12b-it* to insert the English segments at natural clause boundaries in the Hindi sentence. Third, a cleaning pass prompts Gemma to produce a more fluent version of each sentence while retaining as many English words as possible; this is run twice, with a stronger prompt applied to any sentences containing no English words after both passes.

All samples are manually reviewed by the native-speaker authors of the paper to verify fluency and semantic consistency.

| Type | Example |
|---|---|
| **Intra-sentential** | Hier soir, elle a regardé 'N'oubliez pas les paroles !' avec sa famille, and they all had so much fun guessing the lyrics together. |
| **Inter-sentential** | J'ai toujours adoré écouter François Valéry à la radio, sa voix me rappelait les soirées de mon adolescence dans les années 70. But honestly, his recent comments about Claude François really surprised me. |

Table 1: Examples of intra- and inter-sentential Fr-En code-switching from **SwitchLingua**.

### 3.2 Experimental Setup

To understand the internal workings of LLMs when processing code-switching segments, we chose to pursue a mechanistic analysis of different open-source LLMs on the constructed datasets.

#### 3.2.1 Methodology Overview

Inspired by (Nie et al., 2025), we wish to investigate whether some specific neurons are activated only for code-switching. We are interested in the behaviour of the model around **switch points**, which we define as the positions in a sample where the language switches from the target language to English and vice-versa. By comparing against the monolingual baselines at the same position as a control condition, we hope to isolate activations that are genuinely switch-specific, and not just the neuron responding to the target-language token. Identifying such neurons could inform targeted interventions, such as steering a model toward more fluent code-switching or suppressing it entirely, without requiring additional fine-tuning or data collection. Our main pipeline to achieve neuron activation analysis is the following:

1. Load datasets and tokenise the samples
2. Assign token-level language labels to the dataset
3. Run a forward pass through models and collect the neuron activations per layer.
4. Store extracted activations for downstream analysis

This main procedure is then modified for each specific experiment detailed in 4.

#### 3.2.2 Implementation Details

We use the **TransformerLens** library to collect post-MLP activations, except in 4.3 where custom PyTorch forward hooks are used to extract intermediate MLP activations and residual stream representations. To avoid model biases in our results, we run a series of experiments on the following models: **Llama-3-8B**, **Qwen2-7B-Instruct**, **Gemma-7b** and **Mistral-7B-v0.1**, spanning different architectures and training corpora. Notably, **Qwen2** includes substantial Chinese-language pre-training data, allowing us to examine whether a model's training language distribution influences the emergence of language-selective neurons.

## 4 Experiments

To fully characterise the internal processing of code-switched text, we run four different mechanistic interpretability experiments detailed below.

### 4.1 Neuron Clustering Analysis (Exp.1)

Experiment 1 uses cluster analysis to examine how neurons encode multilingual signals more broadly.

To analyse how multilingual signals are encoded, neuron activations were extracted from four models (see Section 3.2.2), across three input conditions: English (baseline), target language

(French or Hindi), and code-switched text. For each neuron, mean activation values were computed under each condition, and two relative shift measures were derived: $\Delta_{\text{target}} = \text{mean}_{\text{target}} - \text{mean}_{\text{english}}$ and $\Delta_{\text{cs}} = \text{mean}_{\text{cs}} - \text{mean}_{\text{english}}$. These metrics quantify how strongly each neuron responds to non-English input relative to the baseline. To focus on the most informative signals and reduce noise from weakly responsive units, only the top $20\%$ of neurons (ranked by absolute activation magnitude) of each model were retained for further analysis.

Clustering was performed independently at three representative depths in each model: the first, middle, and final layer, allowing comparison of early, intermediate, and late-stage representations. For each selected neuron, a feature vector consisting of $(\text{mean}_{\text{target}}, \text{mean}_{\text{cs}}, \Delta_{\text{target}}, \Delta_{\text{cs}})$ was constructed, and hierarchical clustering with Ward linkage, a clustering technique that merges clusters to minimise within-cluster variance, was applied to group neurons with similar response magnitudes and patterns, producing interpretable clusters. The number of clusters was determined automatically using the silhouette score, ensuring a data-driven selection consistent across models and layers. This framework enables systematic comparison of neuron behaviour under code-switched and monolingual conditions across network depth.

### 4.2 Neuron Activation Patterns at Code-Switch Points (Exp.2)

This experiment seeks to isolate neurons that respond specifically to language transitions, rather than neurons merely associated with one language in isolation. For each matched group of `code_switched`, `english` and `target_language` samples, each word is assigned a language label using a vocabulary-overlap heuristic: a word is labelled as English if it appears only in the English baseline, and as target-language if it appears only in the target-language baseline. This word-level boundary is then mapped back to the token space by taking the first token of the switched word as the anchor position.

For a given sample, layer, neuron, and offset $\{-1, 0, +1\}$, a neuron is defined to be *switch-specific* if its post-MLP activation in the code-switched sentence is strictly greater than its activation in every monolingual baseline at the corresponding aligned position:

$$\text{switch-specific}^{(i)}_{n,l,\delta} = \mathbf{1}\left[a^{\text{cs},(i)}_{n,l,\delta} > a^{b,(i)}_{n,l,\delta} \;\; \forall\, b \in \mathcal{B}\right]. \tag{1}$$

We then aggregate over all valid switch events to compute a *consistency fraction* for each neuron $n$ at layer $l$ and offset $\delta$:

$$\text{consistency}_{n,l,\delta} = \frac{1}{N_\delta} \sum_{i=1}^{N_\delta} \text{switch-specific}^{(i)}_{n,l,\delta}, \tag{2}$$

where $N_\delta$ is the number of valid events contributing to offset $\delta$. The resulting score lies in $[0, 1]$, with larger values indicating more reliable switch-specific activation across matched examples. Neurons whose consistency exceeds a threshold $\tau$ (default $\tau = 0.7$) are retained as *consistent switch-specific neurons*.

### 4.3 Cross-Lingual Comparison (Exp.3)

This experiment investigates whether the two CS conditions (Hi-En vs. Fr-En) tend to recruit similar or distinct internal circuits across four analyses.

**Whole-Sentence activation similarity:** For each layer, MLP activations are mean-pooled across tokens to produce a per-condition vector. Jaccard and cosine similarity are then computed between *cs_fr vs. French, cs_fr vs. English, cs_hi vs. Hindi, cs_hi vs. English,* and *cs_fr vs. cs_hi.* Higher similarity to the matrix language than to English indicates matrix-language-dominated processing, while cross-CS similarity assesses whether switching conditions share or diverge in their representations.

**Token-ratio control:** To control for the effect of token frequency, the similarity analysis is repeated on samples with a 45–55% matrix-language token ratio. Persistent matrix-language dominance in this balanced subset suggests the effect is not driven by token frequency alone.

**English-token contextual analysis:** This analysis isolates English tokens within CS sentences and examines whether they activate more like monolingual English or the surrounding matrix language, and whether this contextual pull differs between the Hi–En and Fr-En conditions. A position-matched variant strengthens this by finding the same token in the monolingual English sentence and comparing its activation directly against the same token in the CS sentence,

reflecting local context rather than whole-sentence averaging.

**Layer-wise divergence from English:** At each layer, the L2 distance is computed between the CS residual stream at switch-point positions and the mean residual stream of the corresponding monolingual English sentence, for both CS conditions. This examines whether Hi-En and Fr-En switching disrupts the model's internal state at the same layers.

### 4.4 Identification of Language–Selective Neurons (Exp.4)

Language-selective neurons can reveal information about the model's internal processing of different languages and scripts. In particular, we seek to find whether the model has language–specific circuitry, or instead relies on a shared, language-agnostic internal representation. To identify these neurons, each code-switched sample is tokenised, and the tokens are then labelled by language. This allows us to identify the switch points. By aggregating the mean token post-MLP activation per neuron and per layer for each language, we can compute the language $selectivity$ index:

$$selectivity = (M_t - M_{eng})/(M_t + M_{eng}) \quad (3)$$

where $M_t$, $M_{eng}$ refer to the mean activation for target-language tokens and English tokens, respectively. The selectivity index lies in the range $[-1, 1]$: values closer to 1 indicate that the neuron is more active for the target language, while values closer to $-1$ indicate higher activity for English. By focusing on selectivity around the switch points, we aim to determine whether these neurons track language transitions.

## 5 Results

### 5.1 Neuron Clustering Analysis (Exp.1)

Across all four models, hierarchical clustering reveals a consistent structure: a small set of neurons (8–3748 depending on model) show elevated activations for non-English input, while the majority show higher activations for English (appendix, Figure 4). This aligns with prior work showing that multilingual capability is concentrated in a sparse subpopulation within an otherwise English-centric network (Tang et al., 2024; Wendler et al., 2024; Zhao et al., 2024).

Crucially, this non-English subpopulation activates at near-identical levels for both code-switched and pure target-language input (e.g.LLaMA Hindi: $\Delta_{cs} = +1.57, \Delta_{target} = +1.62$), suggesting that target-language tokens alone are sufficient to recruit the same neural machinery as monolingual input. Models do not appear to represent code-switching as categorically distinct from target-language input.

Some variation exists across languages and models. French conditions produce more clusters, often with negative $\Delta_{target}$ and $\Delta_{cs}$ in the final layer, likely due to script and lexical overlap with English, whereas Hindi consistently yields two clusters with strong positive shifts (see Appendix 4), reflecting its greater representational distance from English.

Qwen stands out for allocating the largest non-English language sensitive clusters as seen in Figure 1, suggesting its broader multilingual training produces more distributed non-English representations rather than the ultra-sparse clusters seen in Gemma (see Appendix 4).

Across all models, non-English activations peak in the final layer, consistent with language-specific neurons concentrating in the topmost layers where semantic content is projected into output tokens (Tang et al., 2024; Zhao et al., 2024).

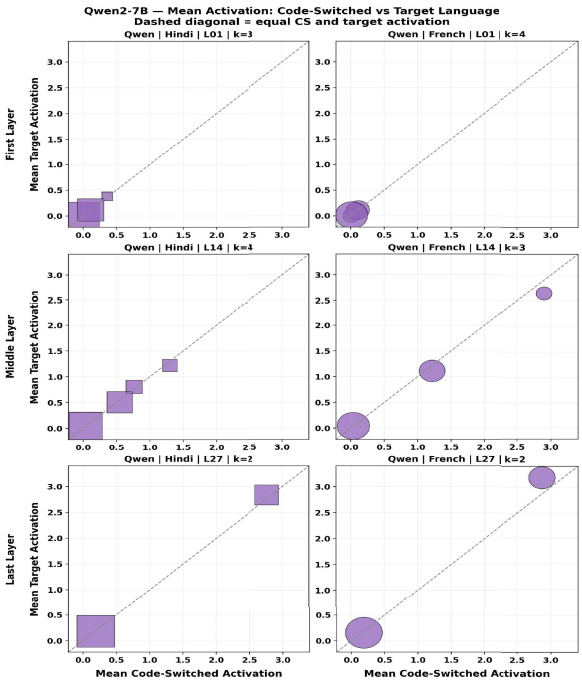


Figure 1: Qwen2-7B Layerwise Clustering

### 5.2 Neuron Activation Patterns at Code–Switch Points (Exp.2)

The results reveal a clear and consistent structural pattern in switch-sensitive activity across all models. Neurons are strongly concentrated at the switch token itself ($\delta = 0$), with counts at the transition point far exceeding those at adjacent offsets ($\delta = \pm 1$) as seen in Table 2. This indicates that reliable switch-specific responses are tightly localised to the switch boundary.

A pronounced asymmetry between language pairs is observed, with Hi–En switches eliciting stronger and more consistent responses than Fr–En. Larger models show the same overall pattern: Qwen2-7B and Mistral-7B exhibit large Hindi advantages ($6.1\times$ and $6.4\times$ respectively), while Gemma-7B and LLaMA-3-8B show more moderate gaps (approximately $1.5\times$). Qwen2-7B produces the largest Hindi-sensitive set overall (490 neurons at $\delta = 0$), substantially exceeding both Gemma-7B (243) and LLaMA-3-8B (133).

Cross-language overlap highlights a generational shift in how models represent switching behaviour. Gemma-7B and LLaMA-3-8B exhibit the highest Jaccard overlap at $\delta = 0$ (0.2121 and 0.2316), suggesting a larger shared, language-agnostic switch-detection component. In contrast, Qwen2-7B shows a lower Jaccard (0.0755) despite 40 shared pairs, reflecting the imbalance between its French and Hindi sets. Overall, overlapping French–English and Hindi–English switch-specific neurons are concentrated primarily in the early-middle to middle layers of each model, suggesting that shared code-switch processing emerges after low-level token encoding but before the final output-oriented layers.

| Model | $\delta$ | Fr | Hi | Ovlp | Jac. |
|---|---|---|---|---|---|
| Mistral-7B | $-1$ | 2 | 0 | 0 | — |
| **Mistral-7B** | **0** | **20** | **128** | **11** | **0.0803** |
| Mistral-7B | $+1$ | 2 | 11 | 0 | — |
| Gemma-7B | $-1$ | 11 | 23 | 1 | 0.0303 |
| **Gemma-7B** | **0** | **157** | **243** | **70** | **0.2121** |
| Gemma-7B | $+1$ | 19 | 5 | 3 | 0.1429 |
| LLaMA-3-8B | $-1$ | 3 | 16 | 0 | — |
| **LLaMA-3-8B** | **0** | **85** | **133** | **41** | **0.2316** |
| LLaMA-3-8B | $+1$ | 4 | 8 | 2 | 0.2000 |
| Qwen2-7B | $-1$ | 2 | 7 | 0 | — |
| **Qwen2-7B** | **0** | **80** | **490** | **40** | **0.0755** |
| Qwen2-7B | $+1$ | 12 | 13 | 1 | 0.0417 |

Table 2: Consistent switch-specific neuron counts and cross-language overlap. Fr/Hi = consistent (layer, neuron) pairs; Ovlp = exact shared pairs between the French–English and Hindi–English sets; Jac. = Jaccard overlap.

### 5.3 Cross–Lingual Comparison (Exp.3)

At the sentence level, both CS conditions align predominantly with their respective matrix languages (the base language that provides the grammatical structure of a sentence) rather than English, but the degree of this alignment differs markedly across language pairs. Fr-En switching shows moderate matrix dominance (cs_fr-French cosine: 0.922-0.937; cs_fr-English: 0.724-0.819), whereas Hi–En switching shows a considerably stronger effect (cs_hi-Hindi: 0.957-0.971; cs_hi-English: 0.317-0.655). The cross-CS similarity (cs_fr vs.cs_hi) is the lowest of any pair across all models, suggesting the two conditions engage largely distinct circuits (Tab 3). This divergence persists in the token-ratio controlled subset (45-55% matrix-language tokens), indicating it reflects something structural about how each language pair is processed rather than a simple frequency artefact.

At the token level, the two conditions diverge in a model-dependent way. For Hi–En CS, Qwen and Mistral show a clear reversal whereby English tokens embedded in cs_hi sentences activate more like Hindi than monolingual English suggesting the Hindi matrix context actively reshapes individual English token representations (Tab 4). No such reversal is observed for Fr–En CS across any model; embedded English tokens show only a mild French-ward pull (0.04-0.05), consistent with the closer representational proximity between French and English. Gemma and LLaMA show this asymmetry between language pairs most clearly, as they preserve English token identity within Fr–En CS but partially lose it within Hi–En CS.

Layer-wise, both conditions share a common peak divergence layer within each model (layer 27 for Qwen and Gemma, layer 31 for LLaMA and Mistral), tentatively suggesting a shared architectural locus for switch-point processing. However, cs_hi consistently produces larger L2 divergence from the English baseline than cs_fr at every layer, with the gap widening in deeper layers (e.g. Mistral: cs_hi 28.5 vs. cs_fr 18.0; LLaMA: 28.7 vs. 21.5). This late-layer amplification suggests that while the two conditions may share early pro-

| Model | *cs_hi* vs. *cs_fr* |
|---|---|
| Qwen2-7B | 0.270 |
| Gemma-7B | 0.506 |
| LLaMA-3-8B | 0.425 |
| Mistral-7B | 0.275 |

Table 3: Cross-pair representational distance between Hi-En and Fr-En CS in late layers. Low cosine similarity indicates the two CS conditions occupy distinct representational spaces with no shared switching mechanism.

| Model | $EN_{hi}$ vs. EN | $EN_{fr}$ vs. EN |
|---|---|---|
| Qwen2-7B | 0.359 | 0.687 |
| Gemma-7B | 0.842 | 0.888 |
| LLaMA-3-8B | 0.730 | 0.865 |
| Mistral-7B | 0.799 | 0.855 |

Table 4: Token-level assimilation (Check 3b) in late layers. $EN_{hi}$ vs. EN: cosine similarity of English token in Hi-En CS context vs. monolingual English. $EN_{fr}$ vs. EN: same for Fr-En CS. Lower values indicate stronger assimilation toward the matrix language. Qwen uniquely shows strong assimilation in Hi-En CS; all models largely preserve English token identity in Fr-En CS.

cessing mechanisms, they increasingly diverge at deeper layers pointing to language-pair-specific rather than fully shared internal circuits.

### 5.4 Identification of Language–Selective Neurons (Exp.4)

Hindi selectivity is consistently stronger than French. At layer 1, Hindi-selective neurons ranged from 36–49% across models, versus only 13–27% for French. This gap likely reflects both script and linguistic differences – Hindi differs from English in script (Devanagari) and language family, while French differs linguistically, but shares the Latin alphabet and much vocabulary. However, we cannot fully disentangle these two factors from our current setup and leave it to future work (see 7).

Language detection is concentrated in early layers. Across all models and both languages, selectivity peaks in layers 0–3 and drops through the middle layers, with a slight recovery at the final layer. In Mistral-7B, early layers show 45–49% Hindi-selective neurons compared to 33–35% in layers 15–20. This is consistent with early layers performing lower-level language identification while deeper layers focus on semantics.

Models differ in selectivity strength. LLaMA-3-8B showed the strongest Hindi selectivity overall, with 65.4% of its 14,336 layer 0 neurons being Hindi-selective, compared to 47.7% of Qwen2-7B's 18,944 neurons and 31.6% of Mistral-7B's 14,336 neurons. The cause of this variation is unclear and may warrant further investigation into how each model's tokeniser, pre-training data, and architecture contribute to language selectivity.

Overlap between Hindi and French selective neurons is low, confirming language-specific circuitry. Jaccard similarity between Hindi-selective and French-selective neuron sets averaged just 0.13–0.19 across models - barely above the $\sim$0.12 expected by chance. The exception is layer 1, where overlap peaks at 0.21–0.27 across all models, suggesting a shared early hub before representations diverge. The absolute selectivity overlap ($\sim$0.31–0.34) is about double target-selective overlap, suggesting shared neurons are broadly selective rather than target-specific, though their function remains unclear. Overall, the data supports dedicated language-specific circuitry as the dominant regime, with a small, shared component concentrated in the earliest layers.

## 6 Discussion

Across all four experiments, a consistent pattern emerges: LLMs do not form a unified representation for code-switched text. Instead, each case aligns with the matrix language, while the embedded language leaves only a brief, local effect at the switch point. This holds across models, though the strength varies by language pair.

**Matrix language dominance:** The most consistent result is that code-switched sentences activate neural machinery similar to monolingual target-language input. Experiment 1 shows this at the neuron level – the same sparse subpopulation of non-English neurons that responds to pure Hindi or French activates at near-identical magnitudes for their code-switched counterparts. Experiment 3 confirms this at the representational level, with CS-to-matrix-language cosine similarity consistently exceeding 0.92 across all models and language pairs. Experiment 2 reveals that the embedded language does leave a detectable trace, but one tightly localised to the switch token itself.

**Asymmetry between Hi-En and Fr-En and its interpretation:** Through all four experiments: Hindi–English code-switching produces stronger matrix dominance, more switch-specific neuron recruitment, greater switch-point divergence from English, and higher language selectivity than

French-English code-switching. Script discontinuity can provide one explanation for this. Moreover, Hindi and English are morphologically different and belong to separate language families. The gap between Hi–En and Fr–En persists in later layers where surface orthographic features have largely been abstracted away, suggesting that linguistic distance contributes independently of script.

**Layer-wise dynamics:** Language-selective neurons are most concentrated in early layers (0–3) across all models and both language pairs (Exp. 4), consistent with lower-level language identification occurring early in the processing pipeline. Despite this, L2 divergence from the English baseline at switch points increases with depth, peaking in the final layers (Exp 3). This suggests that while language identity forms early, its effects build over time. By the final layers, CS representations are most strongly shaped by the matrix-language context. This late-layer divergence is consistently larger for Hi-En than Fr-En.

**Absence of a unified code-switching circuit:** The very low representational similarity between the two CS conditions themselves (late cosine: 0.270-0.506) argues against a language-agnostic code-switching module. Instead, the model appears to handle CS by assimilating each condition into the dominant language's representational framework. Low overlap in switch-specific neurons supports this, though slightly higher overlap in Gemma and Llama suggests some models may share a small switching component.

**Model-level variation:** Qwen shows the strongest matrix dominance. Indeed, embedded English in HindiEnglish CS behaves more like Hindi in later layers, likely due to its broader multilingual training. Gemma shows the weakest dominance and highest cross-CS similarity, suggesting less distinct language-specific circuits. Overall, these findings suggest that matrix-language dominance in CS is not inherent to transformers but learned from training data and design.

## 7 Limitations and Future Work

First, the imbalance in pre-training data representation between French and Hindi across models makes it difficult to disentangle the effects of linguistic and orthographic distance from those of representational familiarity. The observed Hi–En vs. Fr–En asymmetry almost certainly reflects both, and cannot be distinguished in this experimental setup. Second, all analyses are correlational: identifying activation patterns associated with code–switching does not establish causally-driven behaviour. Third, the code–switched datasets used are synthetic and may not capture the distributional properties of real–world code–switching. Fourth, as English is the embedded language we are limited in what can be said about non-English code-switching configurations. Finally, the analyses focus primarily on MLP activations, leaving the role of attention mechanisms at language switch boundaries largely unexamined.

These findings suggest several future directions. The confound between script and linguistic distance could be tested using romanised Hindi, which preserves grammatical structure while removing Devanagari script. Activation patching would allow direct causal testing of whether the identified matrix-language-dominant components are necessary for the observed processing patterns. Extending the analysis to a broader range of language pairs, including non-English matrix languages, lower–resource languages, and typologically diverse pairs , would generalise these findings. Complementing the MLP–focused analyses with systematic attention head analysis is another natural extension.

## 8 Conclusion

In this study we investigate whether code-switching triggers distinct internal behaviour in multilingual LLMs. By constructing synthetic code-switched datasets spanning two linguistic families (Hindi and French) and evaluating multiple transformer models, we find that, despite differing cross-lingual behaviour, models assimilate code-switched text into the representational framework of the matrix language. In some cases, the dominant language even influences how embedded English tokens are encoded. The absence of a shared cross-condition switching circuit suggests that code-switching is not handled as a distinct task for these models. However, mechanistic interpretability remains a developing field, and our findings only capture correlational patterns. We encourage further work (see Section 7) in this direction, as it could enhance embedded-language-sensitive tasks including cross-lingual retrieval and code-switching generation.

## References


Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia. Association for Computational Linguistics.

Amr Mohamed, Yang Zhang, Michalis Vazirgiannis, and Guokan Shang. 2025. Lost in the mix: Evaluating llm understanding of code-switched text.

Ercong Nie, Helmut Schmid, and Hinrich Schuetze. 2025. Mechanistic understanding and mitigation of language confusion in English-centric large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 690–706, Suzhou, China. Association for Computational Linguistics.

Juhyun Oh, Haneul Yoo, and Alice Oh. 2025. Evaluating LLMs' language confusion in code-switching context. In *NeurIPS 2025 Workshop on Evaluating the Evolving LLM Lifecycle: Benchmarks, Emergent Abilities, and Scaling*.

Shana Poplack. 1980. Sometimes ill start a sentence in spanish y termino en espaÑol: toward a typology of code-switching 1. *Linguistics*, 18:581–618.

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.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in English? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394, Bangkok, Thailand. Association for Computational Linguistics.

Peng Xie, Xingyuan Liu, Yequan Bie, Tsz Wai Chan, Yangqiu Song, Yang Wang, Hao Chen, and Kani Chen. 2025. Switchlingua: The first large-scale multilingual and multi-ethnic code-switching dataset. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Yuemei Xu, Ling Hu, Jiayi Zhao, Zihan Qiu, Kexin Xu, Yuqi Ye, and Hanwen Gu. 2025. A survey on multilingual large language models: corpora, alignment, and bias. *Frontiers of Computer Science*, 19(11).

Zheng-Xin Yong, Ruochen Zhang, Jessica Zosa Forde, Skyler Wang, Arjun Subramonian, Holy Lovenia, Samuel Cahyawijaya, Genta Indra Winata, Lintang Sutawika, Jan Christian Blaise Cruz, Yin Lin Tan, Long Phan, Rowena Garcia, Thamar Solorio, and Alham Fikri Aji. 2023. Prompting multilingual large language models to generate code-mixed texts: The case of south East Asian languages. In *Proceedings of the 6th Workshop on Computational Approaches to Linguistic Code-Switching*, pages 43–63, Singapore. Association for Computational Linguistics.

Hongchuan Zeng, Senyu Han, Lu Chen, and Kai Yu. 2025. Converging to a lingua franca: Evolution of linguistic regions and semantics alignment in multilingual large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10602–10617, Abu Dhabi, UAE. Association for Computational Linguistics.

Ruochen Zhang, Samuel Cahyawijaya, Jan Christian Blaise Cruz, Genta Winata, and Alham Fikri Aji. 2023. Multilingual large language models are not (yet) code-switchers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12567–12582, Singapore. Association for Computational Linguistics.

Yiran Zhao, Wenxuan Zhang, Guizhen Chen, Kenji Kawaguchi, and Lidong Bing. 2024. How

do large language models handle multilingualism? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

# A Appendix

Figure 2: First layer

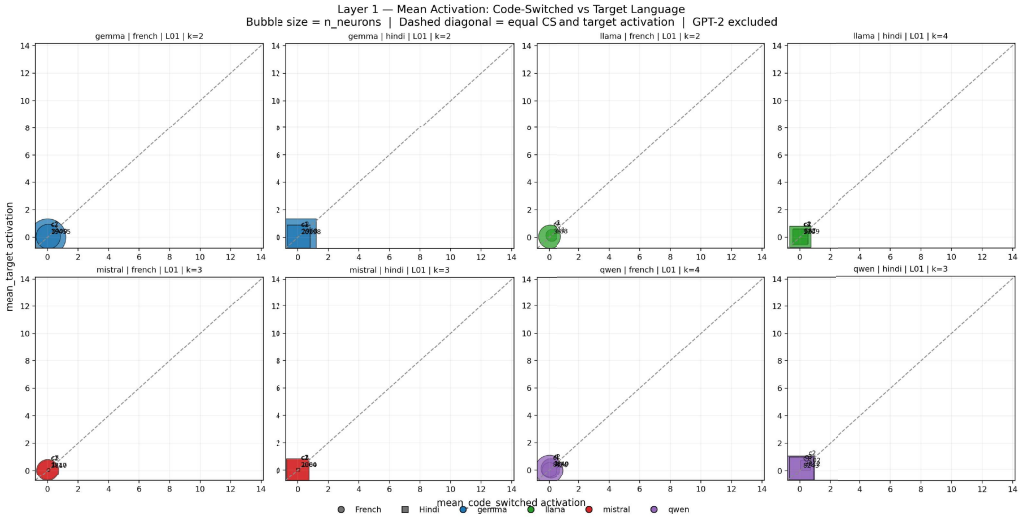


Figure 3: Middle layer

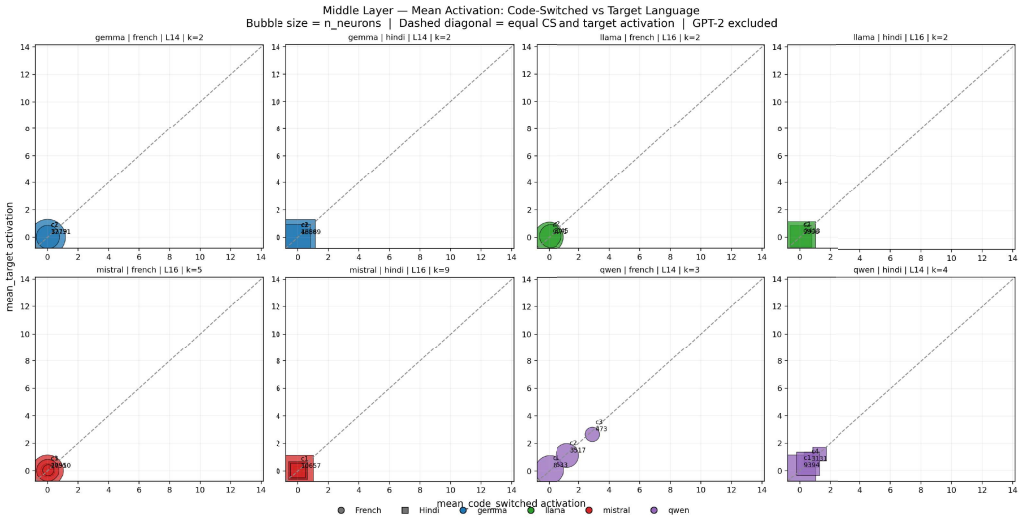


Figure 4: Final layer

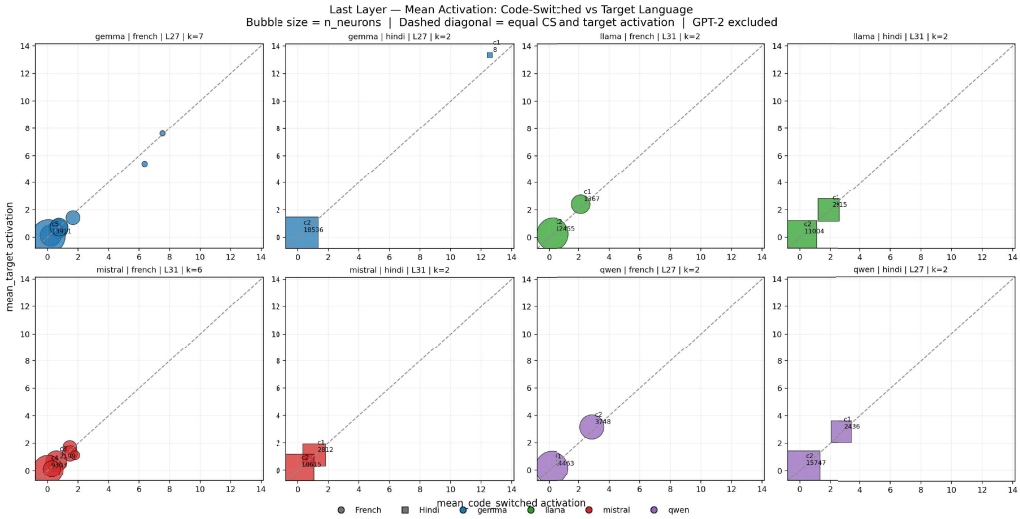


Figure 5: Cluster centroids across network depth (first, middle, and final layers).

| Language | Layer | Model | $Cluster_n$ | $mean_{\text{en}}$ | $mean_{\text{cs}}$ | $mean_{\text{tgt}}$ | $\Delta_{\text{cs}}$ | $\Delta_{\text{tgt}}$ |
|---|---|---|---|---|---|---|---|---|
| *Hindi — First Layer* | | | | | | | | |
| | | Gemma-7B | 20298 | 0.006 | 0.005 | 0.005 | -0.001 | -0.001 |
| | | | 2968 | 0.036 | 0.055 | 0.058 | 0.019 | 0.022 |
| | | LLaMA-3-8B | 2249 | 0.066 | 0.033 | 0.030 | -0.033 | -0.036 |
| | | | 587 | 0.129 | 0.062 | 0.057 | -0.067 | -0.072 |
| | | Mistral-7B | 2900 | 0.004 | 0.001 | 0.001 | -0.003 | -0.003 |
| | | | 2064 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| | | Qwen2-7B | 8243 | 0.022 | 0.006 | 0.005 | -0.016 | -0.016 |
| | | | 3812 | 0.057 | 0.113 | 0.117 | 0.056 | 0.059 |
| *Hindi — Middle Layer* | | | | | | | | |
| | | Gemma-7B | 16589 | 0.001 | 0.001 | 0.001 | -0.000 | 0.000 |
| | | | 4386 | 0.019 | 0.013 | 0.012 | -0.006 | -0.007 |
| | | LLaMA-3-8B | 9996 | 0.007 | 0.007 | 0.008 | 0.000 | 0.001 |
| | | | 2413 | 0.136 | 0.069 | 0.063 | -0.067 | -0.073 |
| | | Mistral-7B | 10657 | 0.001 | 0.000 | 0.000 | -0.001 | -0.001 |
| | | | 1341 | 0.011 | 0.002 | 0.002 | -0.009 | -0.009 |
| | | Qwen2-7B | 9394 | 0.049 | 0.039 | 0.043 | -0.010 | -0.007 |
| | | | 3131 | 1.687 | 0.549 | 0.506 | -1.139 | -1.182 |
| *Hindi — Last Layer* | | | | | | | | |
| | | Gemma-7B | 18536 | 0.227 | 0.164 | 0.156 | -0.063 | -0.071 |
| | | | 8 | 5.707 | 12.573 | 13.348 | 6.866 | 7.641 |
| | | LLaMA-3-8B | 11004 | 0.451 | 0.066 | 0.051 | -0.385 | -0.399 |
| | | | 2915 | 0.360 | 1.929 | 1.980 | 1.568 | 1.619 |
| | | Mistral-7B | 10615 | 0.249 | 0.037 | 0.026 | -0.211 | -0.222 |
| | | | 2812 | 0.380 | 1.076 | 1.105 | 0.696 | 0.725 |
| | | Qwen2-7B | 15747 | 0.745 | 0.191 | 0.183 | -0.554 | -0.562 |
| | | | 2436 | 0.765 | 2.766 | 2.843 | 2.001 | 2.078 |
| *French — First Layer* | | | | | | | | |
| | | Gemma-7B | 19455 | 0.005 | 0.004 | 0.004 | -0.001 | -0.001 |
| | | | 3900 | 0.033 | 0.043 | 0.046 | 0.010 | 0.012 |
| | | LLaMA-3-8B | 3188 | 0.062 | 0.050 | 0.046 | -0.012 | -0.016 |
| | | | 211 | 0.188 | 0.151 | 0.137 | -0.037 | -0.051 |
| | | Mistral-7B | 2710 | 0.005 | 0.004 | 0.003 | -0.001 | -0.001 |
| | | | 1847 | 0.000 | 0.000 | 0.000 | -0.000 | -0.000 |
| | | Qwen2-7B | 8324 | 0.013 | 0.012 | 0.011 | -0.002 | -0.002 |
| | | | 1849 | 0.114 | 0.114 | 0.115 | -0.000 | 0.001 |
| *French — Middle Layer* | | | | | | | | |
| | | Gemma-7B | 17731 | 0.002 | 0.002 | 0.002 | -0.000 | -0.000 |
| | | | 3219 | 0.022 | 0.020 | 0.019 | -0.003 | -0.004 |
| | | LLaMA-3-8B | 9171 | 0.003 | 0.003 | 0.004 | -0.000 | 0.000 |
| | | | 3045 | 0.122 | 0.100 | 0.093 | -0.022 | -0.030 |
| | | Mistral-7B | 10910 | 0.001 | 0.001 | 0.001 | -0.000 | -0.000 |
| | | | 2735 | 0.012 | 0.009 | 0.008 | -0.002 | -0.003 |
| | | Qwen2-7B | 8633 | 0.030 | 0.038 | 0.041 | 0.008 | 0.012 |
| | | | 3517 | 1.516 | 1.216 | 1.110 | -0.300 | -0.406 |
| *French — Last Layer* | | | | | | | | |
| | | Gemma-7B | 13911 | 0.041 | 0.052 | 0.056 | 0.011 | 0.015 |
| | | | 2253 | 0.478 | 0.208 | 0.123 | -0.270 | -0.355 |
| | | LLaMA-3-8B | 12455 | 0.402 | 0.248 | 0.226 | -0.155 | -0.176 |
| | | | 1367 | 0.734 | 2.138 | 2.395 | 1.404 | 1.661 |
| | | Mistral-7B | 9303 | 0.070 | 0.025 | 0.023 | -0.046 | -0.047 |
| | | | 2199 | 0.287 | 0.574 | 0.682 | 0.287 | 0.395 |
| | | Qwen2-7B | 14463 | 0.610 | 0.188 | 0.155 | -0.422 | -0.456 |
| | | | 3748 | 1.271 | 2.868 | 3.179 | 1.597 | 1.909 |

Table 5: Top-2 clusters by neuron count per model, language, and layer. Columns show cluster size and mean activations under each condition.

| Analysis | Condition | Qwen2-7B | Gemma-7B | LLaMA-3-8B | Mistral-7B |
|---|---|---|---|---|---|
| Sentence-level similarity (Mean Cos) | *cs_fr* vs. French | 0.930 | 0.933 | 0.937 | 0.922 |
| | *cs_fr* vs. English | 0.797 | 0.819 | 0.785 | 0.724 |
| | *cs_hi* vs. Hindi | 0.965 | 0.957 | 0.963 | 0.971 |
| | *cs_hi* vs. English | 0.407 | 0.655 | 0.508 | 0.317 |
| | *cs_fr* vs. *cs_hi* | 0.487 | 0.694 | 0.565 | 0.407 |
| Sentence-level similarity (Early Cos) | *cs_fr* vs. French | 0.964 | 0.929 | 0.958 | 0.959 |
| | *cs_fr* vs. English | 0.881 | 0.763 | 0.870 | 0.840 |
| | *cs_hi* vs. Hindi | 0.990 | 0.965 | 0.978 | 0.990 |
| | *cs_hi* vs. English | 0.659 | 0.620 | 0.711 | 0.624 |
| | *cs_fr* vs. *cs_hi* | 0.726 | 0.681 | 0.770 | 0.706 |
| Sentence-level similarity (Late Cos) | *cs_fr* vs. French | 0.953 | 0.938 | 0.945 | 0.917 |
| | *cs_fr* vs. English | 0.667 | 0.741 | 0.733 | 0.619 |
| | *cs_hi* vs. Hindi | 0.985 | 0.972 | 0.981 | 0.988 |
| | *cs_hi* vs. English | 0.199 | 0.479 | 0.429 | 0.231 |
| | *cs_fr* vs. *cs_hi* | 0.270 | 0.506 | 0.425 | 0.275 |
| Embedded EN tokens (Mean Cos) | EN in *cs_hi* vs. English | 0.383 | 0.469 | 0.583 | 0.340 |
| | EN in *cs_hi* vs. Hindi | 0.588 | 0.418 | 0.463 | 0.724 |
| | EN in *cs_fr* vs. English | 0.723 | 0.715 | 0.750 | 0.703 |
| | EN in *cs_fr* vs. French | 0.763 | 0.691 | 0.702 | 0.651 |
| Embedded EN tokens (Early Cos) | EN in *cs_hi* vs. English | 0.590 | 0.599 | 0.700 | 0.602 |
| | EN in *cs_hi* vs. Hindi | 0.820 | 0.439 | 0.572 | 0.705 |
| | EN in *cs_fr* vs. English | 0.826 | 0.719 | 0.809 | 0.795 |
| | EN in *cs_fr* vs. French | 0.844 | 0.619 | 0.754 | 0.731 |
| Embedded EN tokens (Late Cos) | EN in *cs_hi* vs. English | 0.188 | 0.532 | 0.621 | 0.239 |
| | EN in *cs_hi* vs. Hindi | 0.608 | 0.391 | 0.523 | 0.839 |
| | EN in *cs_fr* vs. English | 0.663 | 0.742 | 0.807 | 0.769 |
| | EN in *cs_fr* vs. French | 0.806 | 0.685 | 0.734 | 0.615 |
| Position-matched EN tokens (Mean Cos) | $EN_{hi}$ vs. EN context | 0.505 | 0.725 | 0.693 | 0.570 |
| | $EN_{hi}$ vs. Hindi mean | 0.559 | 0.386 | 0.448 | 0.330 |
| | $EN_{fr}$ vs. EN context | 0.741 | 0.829 | 0.828 | 0.797 |
| | $EN_{fr}$ vs. French mean | 0.718 | 0.624 | 0.659 | 0.608 |
| Position-matched EN tokens (Early Cos) | $EN_{hi}$ vs. EN context | 0.594 | 0.828 | 0.858 | 0.791 |
| | $EN_{hi}$ vs. Hindi mean | 0.804 | 0.421 | 0.564 | 0.476 |
| | $EN_{fr}$ vs. EN context | 0.805 | 0.871 | 0.906 | 0.888 |
| | $EN_{fr}$ vs. French mean | 0.812 | 0.572 | 0.720 | 0.700 |
| Position-matched EN tokens (Late Cos) | $EN_{hi}$ vs. EN context | 0.359 | 0.842 | 0.730 | 0.799 |
| | $EN_{hi}$ vs. Hindi mean | 0.574 | 0.357 | 0.503 | 0.311 |
| | $EN_{fr}$ vs. EN context | 0.687 | 0.889 | 0.865 | 0.855 |
| | $EN_{fr}$ vs. French mean | 0.754 | 0.612 | 0.688 | 0.563 |
| Switch-point divergence (L2 from English) | *cs_fr* mean | 200.4 | 13.5 | 11.5 | 8.4 |
| | *cs_fr* late | 278.1 | 24.5 | 21.5 | 18.0 |
| | *cs_hi* mean | 218.1 | 14.6 | 13.1 | 10.8 |
| | *cs_hi* late | 352.8 | 29.8 | 28.7 | 28.5 |

Table 6: Full results for Exp. 3 (Cross-Lingual Comparison) across all analyses, conditions, and models. Early = mean over layers $\leq$ 3, Late = mean over last 3 layers. Position-matched EN tokens (Check 3b) compare the same English word in CS context vs. monolingual English at identical token positions. Switch-point divergence measures L2 distance between CS and English residual stream at language switch positions.