# Language steering in latent space to mitigate unintended code-switching

Anonymous ACL submission

## Abstract

Multilingual Large Language Models (LLMs) often exhibit unintended code-switching, reducing reliability in downstream tasks. We propose latent-space language steering, a lightweight inference-time method that identifies language directions via PCA on parallel translations and steers token embeddings along these axes to control language identity. Our approach mitigates code-switching while preserving semantics with negligible computational overhead and requires only minimal parallel data for calibration. Empirically, we achieve 95-99% language classification accuracy using a single principal component and reduce next-token distributional divergence by up to 42% across multiple language pairs on Qwen2.5 and Llama-3.2 models. We further analyze the layer-wise evolution of language representations, revealing that language identity concentrates in final layers with near-perfect linear separability. Code and data are released for reproducibility[1].

## 1 Introduction

Multilingual large language models (LLMs) frequently exhibit *unintended code-switching* - generating tokens in languages other than the target despite explicit monolingual instructions. This undermines reliability in production systems and complicates evaluation. Existing solutions require costly fine-tuning or introduce latency via decoding-time interventions like vocabulary masking or classifier guidance.

We propose **latent space language steering**, a cheap inference-time method that exploits low-dimensional *language directions* in hidden representations. By applying PCA to parallel translations, we identify linear subspaces separating languages while preserving semantics. Steering via simple projection (one dot product and vector subtraction per token) mitigates code-switching with negligible overhead.

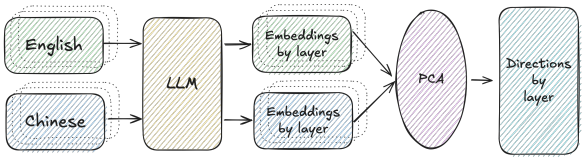



Figure 1: Language steering in hidden space: 1. Collect parallel translations of the same concepts; 2. LLM forward pass to get embeddings by layer; 3. Perform PCA by layer on the merged embeddings.

Our analysis shows language identity concentrates in final layers, with PC1 capturing most of the variance and enabling near-perfect language classification. We validate across Qwen2.5 and Llama-3.2 models on multiple language pairs (English, Spanish, Russian, Chinese), demonstrating reduced distributional divergence while preserving fluency.

**Contributions.** We characterize geometric structure of language identity in multilingual LLMs, introduce a cheap steering method with negligible overhead requiring only parallel translations for calibration, and empirically validate across models and language pairs, showing reduced code-switching while maintaining semantic fidelity.

## 2 Related works

**Code-switching and multilingual control.** While code-switching has been studied for mixed-language inputs in speech and NER (Aguilar et al., 2020; Bali et al., 2014), recent work identifies *unintended* code-switching in multilingual LLM outputs (Ryan et al., 2024; Yoo et al., 2024), degrading user experience and evaluation. Standard mitigations require costly fine-tuning on monolingual data, reinforcement learning with language-specific rewards, or brittle prompt engineering.

[1] https://anonymous.4open.science/r/language-steering-in-latent-space-F14E/

**Language structure in representations.** Work on mBERT and XLM-R showed language identity concentrates in few principal components, separable from semantics in parallel texts (Conneau et al., 2020; Chi et al., 2020). Yang et al. (Yang et al., 2021) removed these components for language-agnostic embeddings. Wendler et al. (Wendler et al., 2024) extended this to decoder-only LLMs, revealing latent language preferences. We exploit this structure for *active generation control*.

**Inference-time control.** Plug-and-play methods steer models without fine-tuning via gradient-based attribute classifiers (Dathathri et al., 2020), future discriminators (Yang and Klein, 2021), or expert-guided decoding (Liu et al., 2021). These typically target sentiment or toxicity with auxiliary models or multiple forward passes. Our approach leverages language's cleaner linear separability for simpler intervention: a single projection per token, no gradients, no auxiliary classifiers.

## 3 Methods

Our approach operates in the latent space of pre-trained multilingual LLMs without fine-tuning. We identify language-specific directions via PCA on parallel translations, then steer representations along these directions to control code-switching.

### 3.1 Language Direction Identification

Given a parallel corpus $\mathcal{D} = \{(s, \ell)\}_{i=1}^{N}$ where $s$ represents semantically equivalent content in language $\ell \in \mathcal{L}$, we extract hidden states $\mathbf{h}_i^{(\ell)} \in \mathbb{R}^d$ from each layer $\ell$ of the LLM. For each layer, we apply PCA to centered embeddings $\mathcal{H}^{(\ell)} = \{\mathbf{h}_i^{(\ell)}\}_{i=1}^{N}$. The first principal component $\mathbf{v}^{(\ell)}$ captures the *language direction*:

$$\mathbf{v}^{(\ell)} = \arg\max_{\|\mathbf{v}\|=1} \sum_{i=1}^{N} \left( \mathbf{v}^\top (\mathbf{h}_i^{(\ell)} - \bar{\mathbf{h}}^{(\ell)}) \right)^2 \quad (1)$$

When semantic content is constant, language identity becomes the dominant variance source, making $\mathbf{v}^{(\ell)}$ an interpretable language axis.

### 3.2 Latent-Space Language Steering

Empirical analysis (Section 4) shows language variance concentrates in final layers. We apply steering only to layers $\ell \geq \ell_{\text{crit}}$. For each token's hidden state $\mathbf{h}_t^{(\ell)}$ during generation, we remove the language component:

$$\tilde{\mathbf{h}}_t^{(\ell)} = \mathbf{h}_t^{(\ell)} - s \left( \mathbf{h}_t^{(\ell)} \cdot \mathbf{v}^{(\ell)} \right) \mathbf{v}^{(\ell)} \quad (2)$$

where $s \in \mathbb{R}^+$ controls intervention strength. This removes language identity while preserving semantic content in the orthogonal subspace, adding only a dot product and vector subtraction per token.

### 3.3 Language Prediction

To validate our approach, we train a logistic regression classifier on final-layer projections $z_i = \mathbf{h}_i^{(L)} \cdot \mathbf{v}^{(L)}$:

$$P(\ell \mid z) = \frac{\exp(w_\ell z + b_\ell)}{\sum_{\ell' \in \mathcal{L}} \exp(w_{\ell'} z + b_{\ell'})} \quad (3)$$

This one-dimensional classifier predicts language identity with high accuracy, demonstrating the interpretability of discovered directions.

## 4 Results

We validate our approach on Qwen2.5-1.5B (Qwen et al., 2025) and Llama-3.2-1B (Grattafiori et al., 2024) open models across multiple language pairs, demonstrating both the existence of interpretable language directions and their effectiveness for code-switching control.

### 4.1 Language direction identification

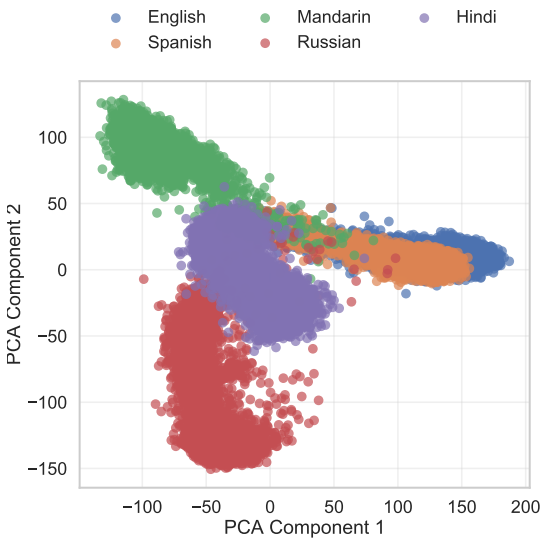
**Experimental setup.** We use Flores Plus (NLLB Team et al., 2024) dataset with 50 parallel samples (English, Spanish, Russian, Chinese, Hindi) for PCA fitting and 100 for validation.

**Language maps reveal emergent clustering.** Figure 2a and 2b show final-layer embeddings projected onto the first two PCs. Despite no explicit language supervision, models spontaneously organize identical content into tight, language-specific clusters with geometrically precise boundaries. Similar language families (Spanish/Russian) cluster closer than typologically distant pairs (English/Chinese), suggesting PC1 captures coarse language distinction while PC2 encodes finer linguistic properties.

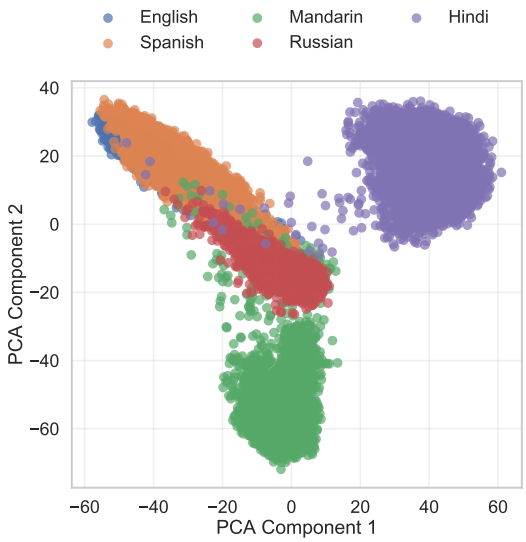
Figures 3a 3b show layer-wise evolution: clusters emerge loosely in early layers and sharpen dramatically in final layers. Figure 3c reveals late-stage specialization for language identity. Pairwise maps (Figures 4a, 4b, 4c, 4d) confirm clean linear separability for all tested pairs.

### 4.2 Language prediction

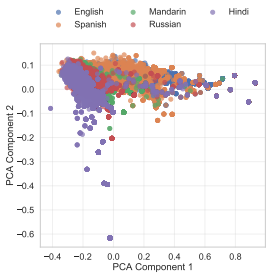
**Experimental setup.** We use Flores Plus (NLLB Team et al., 2024) dataset with 50 parallel samples (English, Spanish, Russian, Chinese, Hindi) for



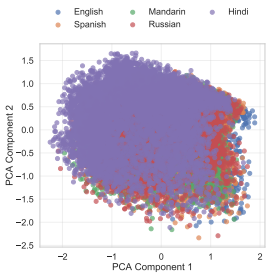

(a) Final layer language map (Qwen)



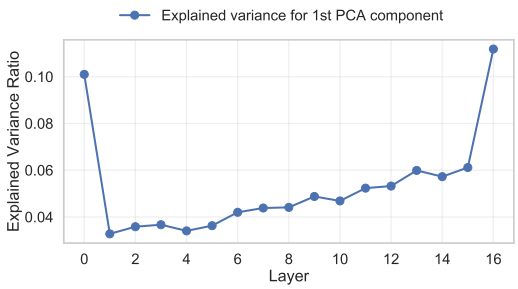

(b) Final layer language map (Llama)




(a) First layer language map (Llama)




(b) Mid-layer language map (Llama)




(c) Explained variance (Llama)

PCA and logistic regression fitting and 100 for validation.

**Classifier shows nearly perfect performance.** Table 1 demonstrates that a simple logistic regression classifier trained only on the first principal component achieves near-perfect accuracy (0.95-0.99) across all language pairs. This validates that language identity information is linearly accessible in the final layer representations, supporting our hypothesis that models maintain explicit language directions in their activation space. We can also clearly see the decreased performance for the English-Spanish pair. It can be attributed to the languages being much more closely related to each other.

| Language pair | Qwen |
|---|---|
| English - Chinese | 0.98 |
| English - Spanish | 0.95 |
| English - Russian | 0.99 |
| English - Hindi | 0.98 |

Table 1: Last layer language classifier accuracy

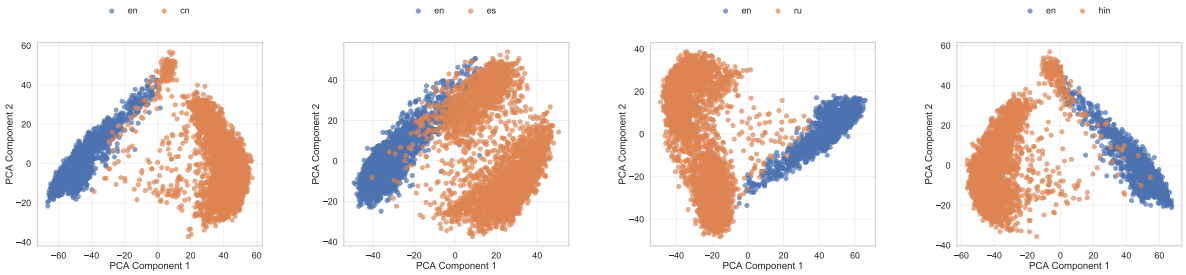
### 4.3 Language steering

**Experimental setup.** We fit the steering module on Flores Plus with 200 parallel samples. We test on 1,000 TED Talks samples where the second half is translated to a target language, creating artificial code-switching. We measure success via KL divergence for the next token prediction distribution (top 100 tokens): $\mathrm{KL}(P_{tokenEN} \| P_{tokenmixed})$ (unsteered baseline) vs. $\mathrm{KL}(P_{tokenEN} \| P_{tokensteered})$. Lower divergence indicates better mitigation.

To find the best steering strength coefficient we conduct a grid search with the best values listed in Table 4 for each language pair.

**Results.** Table 2 demonstrates that steering shifts predictions from Spanish tokens (muy, bastante, tan) to their English equivalents (very, quite, so) while preserving semantic coherence. It changing *how* the model speaks, not *what* it says.

Table 3 shows steering reduces divergence by




(a) English - Chinese (b) English - Spanish (c) English - Russian (d) English - Hindi

| Original | Unsteered | Steered |
|---|---|---|
| still | muy | very |
| going | un | surprised |
| not | bastante | pretty |
| so | tan | starting |
| a | sor | working |
| just | más | content |
| really | a | rather |
| in | empez | impressed |
| very | de | quite |
| sitting | content | – |
| quite | en | great |
| pretty | enc | beginning |
| here | aquí | answering |
| thinking | pens | almost |
| looking | realmente | fully |
| kind | dese | ____ |
| getting | completamente | __ |
| having | feliz | even |
| actually | org | heart |
| surprised | seguro | trying |

Table 2: Top 20 token sample

20% on average. It hints that while we can mitigate the code-switching, the complex structure of the manifold does not allow us to reconstruct the original concept with a simple linear transformation. At the same time, we see that the method shows superior performance for English-Chines language pair. We hypothesize that the root cause is the tokenization and training data imbalance. Most likely, Qwen model prcossed much more English and Chinese data compared to other language pairs. It leads to more granular tokenization and better internal generalization of the concepts.

| Language pair | Qwen |
|---|---|
| English - Chinese | 8.94 → 5.19 |
| English - Spanish | 6.37 → 5.86 |
| English - Russian | 7.78 → 5.43 |
| English - Hindi | 8.73 → 8.73 |

Table 3: KL divergence (no steering → steering)

| Language pair | Qwen |
|---|---|
| English - Chinese | -2.9 |
| English - Spanish | -1.4 |
| English - Russian | -2.2 |
| English - Hindi | 0 |

Table 4: Best steering strength coefficient

## 5 Conclusion

We introduced latent-space language steering, a lightweight inference-time method that mitigates unintended code-switching in multilingual LLMs without fine-tuning. By extracting language directions via PCA on parallel translations, our approach enables precise control over language identity with negligible overhead—one dot product and vector subtraction per token.

Our empirical analysis across Qwen2.5 and Llama-3.2 models reveals that language identity concentrates in final layers with near-perfect linear separability. This geometric structure enables both accurate language classification (95-99% accuracy) and effective steering that manages to restore the original language and reduces distributional divergence by up to 42% while maintaining semantic coherence. The layer-wise evolution shows language identity crystallizes in final layers while early layers focus on semantics, suggesting these occupy largely orthogonal subspaces.

While performance is strongest for typologically distant pairs (English-Chinese), the 20% average divergence reduction indicates that simple linear transformations cannot fully reconstruct monolingual distributions. Our approach demonstrates that interpretable control mechanisms can be extracted from learned representations, offering a practical alternative to expensive fine-tuning for production multilingual systems.

## Limitations

- **Model scale and diversity.** Experiments limited to small models (1-1.5B parameters) and two architectures (Qwen2.5, Llama-3.2). Larger models and diverse architectures remain unexplored.
- **Incomplete reconstruction.** Average 20% KL divergence reduction indicates linear projections cannot fully reconstruct monolingual distributions, suggesting non-linear language-semantic interactions at the manifold level.
- **Dataset coverage.** Evaluation relies on Flores Plus and TED Talks (formal text). Conversational data, technical domains, and naturally occurring code-switching patterns are not tested. Artificial code-switching setup may not reflect real-world scenarios.
- **Language typology.** Limited to Indo-European and Chinese languages. Low-resource languages, right-to-left scripts, and typologically distant languages (agglutinative, tonal) unexplored. Hindi showed no improvement, suggesting tokenization or data balance issues.
- **Layer-wise steering.** Steering applied only to final layers based on empirical observation. Systematic exploration of multi-layer or adaptive steering strategies not conducted.
- **Evaluation metrics.** Primary reliance on KL divergence measures statistical similarity but not semantic preservation, fluency, or downstream task performance. Human evaluation and task-specific metrics needed.
- **Steering calibration.** Optimal coefficients require manual grid search per language pair. Principled or adaptive selection methods not developed.
- **Efficiency benchmarking.** No rigorous latency profiling across hardware configurations or batch sizes to validate negligible overhead claims for production deployment.

## References


Gustavo Aguilar, Sudipta Kar, and Thamar Solorio. 2020. Lince: A centralized benchmark for linguistic code-switching evaluation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1803–1813, Marseille, France. European Language Resources Association.

Kalika Bali, Jatin Sharma, Monojit Choudhury, and Yogarshi Vyas. 2014. "i am borrowing ya mixing ?" an analysis of English-Hindi code mixing in Facebook. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 116–126, Doha, Qatar. Association for Computational Linguistics.

Ethan A. Chi, John Hewitt, and Christopher D. Manning. 2020. Finding universal grammatical relations in multilingual BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5564–5577, Online. Association for Computational Linguistics.

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online. Association for Computational Linguistics.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. *Preprint*, arXiv:1912.02164.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021. Dexperts: Decoding-time controlled text generation with experts and anti-experts. *Preprint*, arXiv:2105.03023.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841–846.

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. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Michael J. Ryan, William Held, and Diyi Yang. 2024. Unintended impacts of LLM alignment on global representation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16121–16140, 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.

Kevin Yang and Dan Klein. 2021. FUDGE: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535, Online. Association for Computational Linguistics.

Ziyi Yang, Yinfei Yang, Daniel Cer, and Eric Darve. 2021. A simple and effective method to eliminate the self language bias in multilingual representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5825–5832, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Haneul Yoo, Yongjin Yang, and Hwaran Lee. 2024. Code-switching red-teaming: LLM evaluation for safety and multilingual understanding. *arXiv preprint arXiv:2406.15481*.