ReCoVeR the Target Language: Language Steering without Sacrificing Task Performance

Hannah Sterz[1] **Fabian David Schmidt**[2] **Goran Glavaš**[2] **Ivan Vulić**[1]
[1]University of Cambridge [2]University of Würzburg

Abstract

As they become increasingly multilingual, Large Language Models (LLMs) exhibit more *language confusion*, i.e., they tend to generate answers in a language different from the language of the prompt or the answer language explicitly requested by the user. In this work, we propose ReCoVeR (REducing language COnfusion in VEctor Representations), a novel lightweight approach for reducing language confusion based on language-specific *steering vectors*. We first isolate language vectors with the help of multi-parallel corpus and then effectively leverage those vectors for effective LLM steering via fixed (i.e., unsupervised) as well as trainable steering functions. Our extensive evaluation, encompassing three benchmarks and 18 languages, shows that ReCoVeR effectively mitigates language confusion in both monolingual and cross-lingual setups while at the same time—and in contrast to prior language steering methods—retaining task performance. Our data code is available at https://github.com/hSterz/recover.

1 Introduction

Large Language Models (LLMs) are becoming increasingly multilingual (Aryabumi et al., 2024; Team et al., 2025), progressively demonstrating more and more of their abilities across a broader set of natural languages. Broadening language support, however, increases the risk of *language confusion* (Marchisio et al., 2024): the phenomenon where an LLM answers in a language that is different from the language that the user—explicitly or implicitly—requested, or switches the language mid-reply. For instance, Marchisio et al. (2024) report GPT4o has a line-level pass rate of 88% for the Cross-LC Portuguese subset of the LCB. Thus, a non-negligible proportion of the generated lines are not in the expected language. While language confusion is not a frequent phenomenon in all languages, it has a detrimental effect on the user

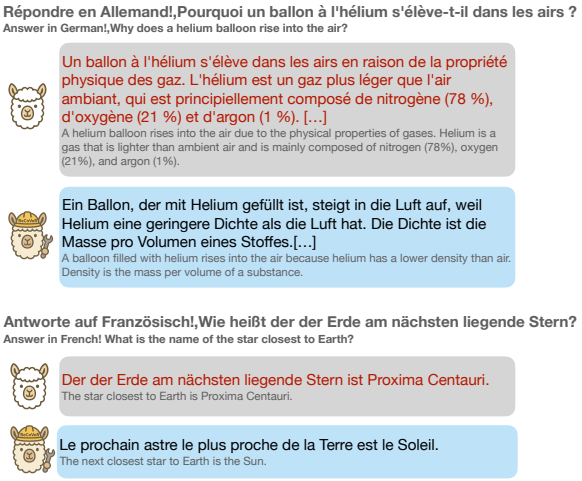


Figure 1: Generated answers for Llama 3.1 without and with ReCoVeR.

experience. Switching to a language that the user did not specify is likely to completely prevent the user from understanding the generated response. An inability to interact with users in their preferred language(s) hinders the global adoption of LLMs and—given that confusion is more prevalent for low-resource languages—exacerbates the exclusivity of state-of-the-art language technology. Such inconsistencies also complicate evaluation, as metrics that assume stable language output may overestimate true model performance.

Intuitively, language confusion can be somewhat mitigated with in-context examples (i.e., few-shot in-context learning, ICL) in the desired response language as well as via multilingual instruction-tuning (Marchisio et al., 2024), but these assume existence of labeled data in each target language of interest. In contrast, Yunfan et al. (2025) propose an unsupervised inference-time intervention to mitigate language confusion based on *language vectors* that are added to hidden representations in the forward pass. While an inference-time solution, this approach suffers from two key drawbacks: (1)

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 19390–19405
November 4-9, 2025

language vectors are computed jointly, i.e., dependent to one another, which means that all language vectors need to be recomputed from scratch when adding a new language; (2) while it improves language fidelity, it actually harms task performance (e.g., answer accuracy in question answering).

Contributions. In this work, we introduce a novel lightweight approach for reducing language confusion with language vectors, dubbed ReCoVeR (REducing language COnfusion in VEctor Representations), illustrated in Figure 2. We use multi-parallel data to pre-compute (1) language vectors for each Transformer layer (as average representations over all in-language input samples and all token positions) and then (2) language-agnostic, content vectors as averages of all language-specific vectors. By subtracting the content vectors from the language vectors, we obtain the language-specific representations (i.e., the *steering vectors*), which finally allow us to mitigate language confusion at inference time by means of simple arithmetic operations. We also experiment with trained interventions that learn how to compute the steering vector from the individual language vectors.

Conceptually, like the concurrent work of Yunfan et al. (2025), ReCoVeR adds (and subtracts) language-specific representations from intermediate token representations, but the way we compute and apply language vectors allows for seamless addition of new languages, without the need to recompute language vectors for all existing languages (see Figure 1 for example generations of Llama). Our extensive evaluation focuses on language confusion in both monolingual and cross-lingual setups and encompasses 18 typologically diverse languages. ReCoVeR effectively reduces language confusion across the board, and crucially—unlike the concurrent approach (Yunfan et al., 2025)—largely retains task performance.

2 Background and Related Work

We first provide a brief overview of the body of work on manipulating LLMs via steering vectors and then describe the existing work on using steering vectors for mitigating language confusion.

Steering Language Models. Hidden representations of LLMs conflate semantic content of the input text (i.e., prompt) with other aspects such as language, script, or style (Bricken et al., 2023); the tokens that the LLM generates are predicted from representations that mix all these aspects. The body of work on *representation engineering* aims to disentangle the contributions of different aspects in hidden representations, in order to accentuate or attenuate an aspect of interest, steering that way the behavior of an LLM in the desired direction (Zou et al., 2023; Turner et al., 2023; Wang et al., 2025).

Representation steering (Stolfo et al., 2025; Stoehr et al., 2024; Zhang and Viteri, 2025; Subramani et al., 2022; Hernandez et al., 2024), as a predominant form of representation engineering, captures the representations of particular concepts (e.g., *truthfulness* or *toxicity*), as encoded in the representation spaces of LLM's layers, and then modifies hidden representations of the input with these concept representations: this has the goal of steering the model behavior in the direction of the concept (e.g., towards more truthful or less toxic generations). The success of representation steering can be explained by the linear representation hypothesis, which posits hidden representations to be linear combinations of aspect/concept vectors (Park et al., 2024). The most common ways of computing concept vectors is (i) by contrasting representations obtained from a set of positive examples which exhibit the desired behavior (e.g., non-toxic generations) against those obtained from a set of negative examples which do not exhibit the desired behavior (e.g., toxic generations) (Jorgensen et al., 2024; Rimsky et al., 2024; Cao et al., 2024) or (ii) by means of linear probing, identifying the most concept-sensitive dimensions of the hidden representations (Alain and Bengio, 2017; Yunfan et al., 2025). The desired behavior is then induced by adding (or subtracted) the concept vectors from from hidden states at inference time (Singh et al., 2024; Liu et al., 2024).

Steering to Mitigate Language Confusion. The expected language of an LLM's response is given either (i) implicitly, where the model is expected to provide the answer in the language of the prompt or (ii) explicitly, with the language of the response specified as part of the prompt (e.g., *'What is the capital of France? Answer in German!'*), i.e., the expected answer language differs from the prompt language. Throughout the paper, we refer to the former as monolingual language confusion (Mono-LC) and denote the latter as cross-lingual language confusion (Cross-LC). Intuitively, state-of-the-art LLMs suffer from language confusion more in cross-lingual than in monolingual setups (Marchisio et al., 2024). As we show in §5, owing to their

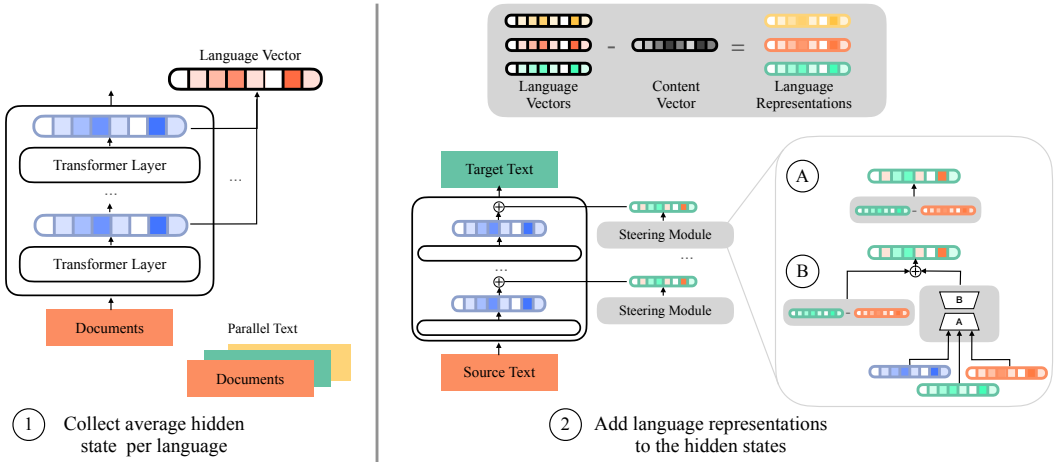


Figure 2: Illustration of ReCoVeR. 1. The language vectors are computed based on the hidden representations on multi-parallel text. 2. The language representations are obtained by subtracting content from language vectors. These language representations can be used in A) unsupervised arithmetic steering and B) learned steering module.

English-centric nature, LLMs especially struggle with Cross-LC when the prompt is not in English.

Language confusion occurs when the model answers in the wrong language or switches between languages during generation (Marchisio et al., 2024). Tang et al. (2024b) find that specific neurons encode the language information, which points to representation steering as a suitable framework for mitigating language confusion: if we can find reliable concept vectors for languages, we could steer the generations towards the target language. Following this intuition, in their language steering at inference (LSI) approach Yunfan et al. (2025) capture language vectors by identifying the most language-sensitive dimensions in the hidden states via linear probing. For each language l, LSI obtains a *representation mask* M_l that describes which dimensions are relevant for correctly predicting l. Next, LSI determines the difference between hidden representations when prompted with an instruction-only target language prompt V_l^* and a prompt with an additional in-context example V_l, following (Marchisio et al., 2024), who report that in-language examples in the context reduce language confusion. V_l is the language mask applied to the hidden state: $V_l = M_l \odot h$. They then compute the language representation (layer i):

$$r_l^{(i)} = \frac{1}{K} \sum_{k=1}^{N} V_{l,k}^{(i)}, -V_{l,k}^{(i)*}. \quad (1)$$

Here, K is the number of prompts. Intuitively, this formula reduces the tendency to answer in the dominant language, i.e., English. Finally, to steer the representations at inference towards the target language, they add $r_l^{(i)}$ (scaled with the hyperparameter γ) to the corresponding hidden representations $h^{(i)}$: $\hat{h}^{(i)} = h^{(i)} + \gamma r_l^{(i)}$.

LSI comes with three prominent drawbacks. First, being based on the language classification probe to produce language-specific masks, LSI produces language vectors r_l that directly depend on the set of languages for which the probe is trained; this prevents an easy post-hoc addition of new target languages as that requires retraining the probe and recomputing r_l vectors for all existing languages too. Second, language representations r_l are computed focusing on the dominant LLM language: this, however, limits the applicability of LSI as it cannot be used for mitigating Cross-LC for non-English prompts. Finally, the reduced language confusion that LSI achieves comes at the cost of a substantial loss of task performance (see §5). We believe that this is because LSI only steers (i.e., changes) hidden representations in a subset of dimensions (i.e., the most language-specific dimensions, as identified by the probe). This makes the steered representations more likely to fall out the output representation distribution of the layer and as such they "confuse" the rest of the model.

3 Lightweight Language Steering

We introduce ReCoVeR (REducing language COnfusion in VEctor Representations), a lightweight language steering approach that addresses the shortcomings of the existing approaches. ReCoVeR uses a (readily available) multi-parallel dataset with a large language coverage to estimate language-specific vectors and, unlike ICL- and fine-tuning-

based mitigation of LC (Marchisio et al., 2024), does not require labeled task data in target languages. ReCoVeR computes language representations in relation to the average representations across all languages: as such, ReCoVeR is able to support Cross-LC with non-English prompts. Importantly, ReCoVeR steering vectors for new languages can easily be computed post-hoc, without the need to recompute the steering vectors of existing languages, which is one of the key drawbacks of LSI (Yunfan et al., 2025).

Steering comprises two sub-problems. The first one is isolating the vector that captures how a concept (in our case, a language) is encoded in LLM's hidden representations (§3.1). The second subproblem is designing the steering strategy, i.e., how to use the concept (language) vectors to manipulate the hidden representations so that the model generates text with the desired properties (in our case, in the desired target language). To this end, we explore both unsupervised (§3.2) and supervised strategies (i.e., learning how to combine language vectors for optimal steering; §3.3).

3.1 Isolating Language Vectors

We start from the following intuitive hypothesis: differences in hidden representations obtained for a pair of parallel texts, i.e., texts that are mutual translations, primarily stem from how the LLM encodes the input language. Because of this, we propose to use a multi-parallel corpus to isolate language representations. For each language $l \in L$ covered by the multi-parallel corpus D_L, we compute one language vector $v_l^{(i)}$ for each transformer layer i (from 1 to N) as the mean of the hidden representations of layer i, averaged across all tokens of all input samples in language l (i.e., across all tokens of D_l, the monolingual portion of D_L):

$$v_l^{(i)} = \frac{1}{|D_l|} \sum_{x \in D_l} \frac{1}{P} \sum_{p=1}^{P} h_p^{(i)}(x)$$

where $h_p^{(i)}$ denotes the hidden representation of the p-th token of the input prompt x at the output of the i-th layer.[1] The obtained language vectors $v_l^{(i)}$, however, still conflate the representation of the language l with the aggregate representation of the "content" of our multi-parallel corpus D_L. In order to obtain language representations $r_l^{(i)}$ that are to be effectively used for language steering, we first need to eliminate the corresponding "content" representations $c^{(i)}$ from respective language vectors $v_l^{(i)}$. To this end, we assume that we will obtain the language-agnostic content vector $c^{(i)}$ by averaging language vectors $v_l^{(i)}$ across all languages $l \in L$:

$$c^{(i)} = \frac{1}{|L|} \sum_{l \in L} v_l^{(i)}$$

We then obtain the language representations by subtracting the content vectors from respective language vectors: $r_l^{(i)} = v_l^{(i)} - c^{(i)}$. We next seek to exploit the language representations r_l computed from the multi-parallel corpus D_L to steer the LLM towards generations in the target language l.

3.2 Unsupervised Language Steering

We are looking to integrate $r_l^{(i)}$ into the hidden representations, output of the i-th layer, in a manner that fulfills two mutually conflicting objectives: (1) we need *enough* information from $r_l^{(i)}$ to steer the generation towards l and at the same time (2) change the hidden representations *as little as possible*, in order to prevent model collapse. We find that, for Mono-LC, L2-normalizing the r_l and scaling it (with a hyperparameter α) yields steering vectors that can achieve both goals:

$$\hat{h}^{(i)} = h^{(i)} + \alpha \frac{r_l^{(i)}}{|r_l^{(i)}|} \tag{2}$$

For Cross-LC, where the prompt is in a *source* language and the expected answer is in a different *target* language, we need an additional steering component to discourage the LLMs from generating the answer in the language of the prompt. To this end, we set our final steering vector to the difference between $r_{target}^{(i)}$ and $r_{source}^{(i)}$, i.e., we steer the representations towards the target language *and* away from the source language (and we again L2-normalize and scale the steering vector):

$$\hat{h}^{(i)} = h^{(i)} + \alpha \frac{r_{target}^{(i)} - r_{source}^{(i)}}{|r_{target}^{(i)} - r_{source}^{(i)}|} \tag{3}$$

Note that, due to the fact that ReCoVeR does not compute language representations r_l relative to any fixed dominant/pivot language, our steering via above language arithmetic applies to arbitrary pairs of source and target languages in Cross-LC.

[1] We exclude the representations of the first token in each sample, i.e., the *'beginning of sequence'* token, assuming that its representation encodes the input language less prominently.

We modify hidden representations $h^{(i)}$ according to Eq. 2 (Mono-LC) and Eq. 3 (Cross-LC) for all tokens in the input sequence except the first token, following our assumption that representations of the sequence start token do not encode language components. We additionally hypothesize that retaining the norm of hidden representations after steering may be important for preventing model collapse. We thus introduce an additional binary-value hyperparameter that decides whether to restore the representation norms after steering, i.e., whether to ensure that $||\hat{h}^{(i)}|| = ||h^{(i)}||$.

3.3 Learning Language Steering

We next investigate if we can learn, in a sample-efficient manner, a steering function that is more effective than our simple unsupervised steering from §3.2. Aiming for sample-efficient training, we choose our steering function to be a low-rank intervention (with a residual), to which we input our language representation(s) $r_l^{(i)}$ concatenated to a hidden representation $h^{(i)}$:

$$h_+^{(i)} = \hat{h}^{(i)} + AB[h^{(i)}; r_{target}^{(i)}; r_{source}^{(i)}]$$

with $A \in \mathbb{R}^{3d \times r}, B \in \mathbb{R}^{r \times d}$ as trainable parameters of the steering function. For training stability, as common in low-rank interventions, we zero-initialize the up-projection B. $\hat{h}^{(i)}$ is defined as:

$$\hat{h}^{(i)} = h^{(i)} + \alpha \frac{r_{target}^{(i)} - \beta r_{source}^{(i)}}{|r_{target}^{(i)} - \beta r_{source}^{(i)}|} \quad (4)$$

The above formula applies to Cross-LC and Mono-LC. To make it compatible with Mono-LC (where $r_{target}^{(i)} = r_{source}^{(i)}$) we introduce parameter β to scale the source representation:

4 Experimental Setup

Obtaining Steering Vectors. To isolate the language representations r_l (§3.1), we need a multi-parallel corpus D_L: here, we resort to FLORES-200 (Costa-Jussà et al., 2022), a sentence-level multi-parallel dataset covering 200 languages.

We additionally require a training dataset for our learnable steering function (§3.3). This dataset should (1) contain both monolingual as well as cross-lingual prompt-response pairs and (2) cover a wide range of typologically diverse languages (and a wide range of tasks), in order to enable cross-lingual generalization of our learned steering function. We obtain such a dataset by first sampling 4400 single-turn instances (i.e., prompt-response pairs) from the instruction-tuning dataset Tulu v3 (Lambert et al., 2024) and then translate each of them (with GPT-4o) by sampling the translation language for the prompt and (independently) for the response from the following set of languages: English, Spanish, French, German, Portuguese, Russian, Chinese, Japanese, Arabic, Hindi, Indonesian, Hebrew, Tamil, Farsi, Thai, Polish, Dutch, Bengali. We increase the likelihood of sampling monolingual pairs to ensure that they are sufficiently represented. We deliberately exclude from translation four other languages present in the evaluation benchmarks (Italian, Korean, Turkish, and Vietnamese) in order to test our learned steering for (zero-shot) cross-lingual generalization.[2] For details on the dataset and translation quality see A.3.

Evaluation Benchmarks. We employ three different benchmarks for measuring language confusion. Two of them come with task-specific annotations, allowing us to also measure the impact of our language steering on task performance.

Language Confusion Benchmark (LCB) (Marchisio et al., 2024) covers 15 languages and serves to measure language confusion. The corresponding metrics are Line-Level Pass Rate (LPR) and Word-Level Pass Rate (WPR). For LPR, the LLM generations are split into lines and each line is classified by the language identification classifier; LPR is then simply the percentage of lines in the requested/expected language. WPR is the proportion of words in the correct lines (identified for LPR) in the correct script[3]. LCB consists of two portions: monolingual (LLM expected to reply in the language of the prompt) and cross-lingual (LLM instructed to reply in the specified language, different from the prompt language).

MultiQ (Holtermann et al., 2024) is a multi-parallel QA dataset covering 137 languages. We use it to evaluate language confusion and QA accuracy[4]. For the latter, as in the original work, we eval-

[2] Our final multilingual/cross-lingual instruction tuning dataset contains 4400 instances covering 18 languages: we provide the number of samples per language in Table 7 and the training details in §A.2.

[3] Determining the language from a single word is challenging, as the same words can occur in multiple languages. The script of the word serves as an approximation. As a result, WPR is only applicable to languages with non-Latin scripts.

[4] We only consider correct answers in the expected language as correct.

uate with LLM-as-a-judge (GPT4o). We select 5 MultiQ languages for the Mono-LC evaluation. We also create cross-lingual instances with non-English prompts by using prompts in German, Basque, Farsi, French, Swahili, Turkish, and Chinese, including the translation of the cross-lingual instruction *'Answer in X!'*, with X as the target language.

CrossSum (Bhattacharjee et al., 2023) is a cross-lingual summarization benchmark. encompassing more than 1500 language pairs. Due to computational constraints, we limit our evaluation to four language pairs: English-Spanish, Spanish-French, French-Turkish, and Turkish-Swahili.

Models. We experiment with three different open multilingual instruction-tuned LLMs of varying size and declared language support: Llama 3.1 (Grattafiori et al., 2024) (8B model; officially supports 7 languages), Qwen 2.5 (Yang et al., 2024) (7B model; 29 languages), and Gemma 2 (Team et al., 2024) (2B; no. languages undeclared).

Besides against the original model (i.e., without any language steering), we compare the performance of ReCoVeR against the steering with LSI (Yunfan et al., 2025) and language-specific neurons(Kojima et al., 2024). For LSI, we obtain the language masks M_l for with samples from WikiLingua (Ladhak et al., 2020) and for languages not covered by WikiLingua, we use texts from Wikipedia[5]. We provide further details on LSI parameters in Table 6. We evaluate our unsupervised steering (§3.2; denoted as ReCoVeR) as well as our learned steering functions (§3.3, denoted as ReCoVeR+).

5 Results and Discussion

We first show and discuss the language confusion and task performance in Mono-LC and Cross-LC setups and then provide further analyses for some of the design dimensions for language steering.

5.1 Language Confusion

Monolingual Language Confusion. The Mono-LC (i.e., where LLMs are expected to answer in the language of the prompt) results on LCB and MultiQ are shown in Table 1 (for detailed, per-language results see Tables 11, 12, and 13). On LCB, Llama and Qwen exhibit robust out-of-the-box performance (>98% LPR), offering limited opportunity for further gains. Gemma's performance is notably weaker: it more often defaults to English, regardless of the prompt language. Steering with LSI (Yunfan et al., 2025) reduces language confusion for all models on both LCB and MultiQ, but at the same time results in large drops of QA accuracy on MultiQ (from -4 percentage points (pp) for Gemma to almost -12 pp for Llama). Our ReCoVeR variants overall mitigate language confusion comparably or better (e.g., +8pp compared to LSI for Gemma on LCB), but in contrast to LSI, our ReCoVeR (+) language steering actually improves the QA performance for Qwen and Gemma. The QA performance gains are particularly prominent with our trained language steering, ReCoVeR+ (+5pp for Qwen and +9pp for Gemma). These results suggest that our language representations are not just effective on their own (ReCoVeR), but also—considering that we train ReCoVeR+ on merely 4400 instances (see 4)—that they enable highly sample-efficient learning of the language steering intervention.

	LCB	MultiQ	
Model	LPR	LPR	Acc
LLama 3.1	98.7	94.5	**64.4**
+ LSI	99.0	95.7	52.8
+ Lang Neuron	**99.2**	-	-
+ ReCoVeR	99.1	93.8	61.3
+ ReCoVeR+	99.1	**95.8**	62.1
Qwen 2.5	98.3	90.7	61.8
+ LSI	98.0	92.7	51.0
+ Lang Neuron	96.2	-	-
+ ReCoVeR	97.7	92.0	62.7
+ ReCoVeR+	**98.5**	**93.2**	**66.5**
Gemma 2	88.4	91.8	38.5
+ LSI	90.2	**93.5**	34.8
+ Lang Neuron	89.7	-	-
+ ReCoVeR	87.8	91.6	38.3
+ ReCoVeR+	**98.1**	92.9	**47.4**

Table 1: Mono-LC results on LCB and MultiQ.

	de	es	id	sw	zh
Llama 3.1	52.8	55.3	50.2	38.3	51.3
+ ReCoVeR	52.8	54.9	50.2	38.6	51.5

Table 2: MMLU accuracy for Llama 3.1 out-of-the-box and with ReCoVeR.

Effect on Task Performance. When steering a model toward a specific language, it is essential to preserve task performance. To assess the impact of language steering, we evaluate MMLU (Hendrycks et al., 2021) translated into German, Spanish, Indonesian, Swahili, and Chinese using Llama 3.1. The model is prompted to output only the correct answer option. As shown in Table 2, ReCoVeR maintains performance across languages: the absolute difference between the baseline (out-of-the-

[5]Obtained from https://huggingface.co/datasets/wikimedia/wikipedia

	Model	ar	de	es	fr	hi	id	it	ja	ko	pt	ru	tr	vi	zh	avg
zero-shot	LLama 3.1	90.4	95.2	95.6	94.0	91.9	89.3	93.9	78.2	90.3	91.9	90.2	95.3	94.6	83.2	91.0
	+ LSI	92.8	96.9	94.9	96.0	96.5	**90.9**	97.0	82.4	92.0	96.7	92.6	92.8	94.8	84.2	92.9
	+ Lang Neuron	94.3	95.2	95.0	95.0	95.3	86.3	95.6	89.5	92.1	92.2	95.6	97.3	98.0	89.6	93.6
	+ ReCoVeR	99.2	**99.7**	**98.6**	**99.7**	99.3	90.1	**98.6**	**97.6**	**100.0**	**97.0**	99.7	**97.9**	**99.0**	**94.8**	**97.9**
	+ReCoVeR+	**99.6**	98.3	98.3	98.7	**100.0**	88.2	96.9	97.3	99.7	94.2	**100.0**	96.6	97.3	93.6	97.0
	Qwen 2.5	93.2	95.5	94.7	93.4	92.5	87.3	93.1	90.1	93.6	88.6	95.4	92.5	91.9	90.9	92.3
	+ LSI	94.4	92.9	92.7	90.8	94.5	86.1	90.3	89.9	92.8	88.5	94.4	91.6	90.1	90.5	91.4
	+ Lang Neuron	94.4	91.1	91.5	96.5	93.0	88.7	95.8	90.7	97.9	90.6	91.8	91.3	95.9	91.8	92.9
	+ ReCoVeR	97.9	97.6	95.9	95.1	96.9	86.0	95.9	94.5	97.0	92.2	98.0	95.3	96.9	**93.0**	95.2
	+ ReCoVeR+	**98.9**	**99.3**	**98.3**	**97.6**	**98.3**	**91.1**	**98.6**	**99.7**	**98.6**	**92.8**	**98.2**	**97.6**	**97.6**	91.3	**97.0**
	Gemma 2	87.0	91.0	91.0	87.0	84.0	71.0	89.0	76.0	77.0	91.0	77.0	83.0	71.0	70.0	81.8
	+ LSI	58.1	78.0	86.2	55.4	84.7	76.4	77.2	81.2	76.2	88.8	93.2	88.8	94.2	77.0	80.0
	+ Lang Neuron	71.2	70.9	81.6	76.9	75.8	64.8	71.8	77.9	83.2	81.0	78.0	71.7	73.0	72.8	75.3
	+ ReCoVeR	90.2	96.8	97.6	96.2	**99.7**	80.4	97.9	94.3	96.5	**95.2**	97.8	94.7	**98.6**	87.4	94.5
	+ ReCoVeR+	**94.4**	**98.6**	**100.0**	**98.6**	99.6	**89.8**	**98.3**	**95.2**	**99.3**	94.2	**98.9**	**97.9**	**98.6**	**88.8**	**96.6**
5-shot	LLama 3.1	94.6	96.6	95.9	94.6	97.6	86.8	96.6	93.6	94.8	92.9	96.6	97.6	96.6	90.2	94.3
	+ LSI	99.3	**99.7**	96.2	97.6	96.4	91.2	99.0	95.0	97.4	**97.3**	99.0	96.7	97.8	93.6	96.9
	+ ReCoVeR	**100.0**	99.6	**99.7**	**98.6**	**100.0**	88.2	**99.3**	**98.6**	**100.0**	95.5	99.2	**98.9**	98.6	**96.6**	**98.1**
	+ ReCoVeR+	98.5	**99.7**	98.3	98.3	99.3	**92.1**	98.3	98.3	**100.0**	93.5	**100.0**	97.1	**99.7**	92.2	97.5
	Qwen 2.5	94.7	95.7	97.2	95.1	93.4	84.5	93.7	89.8	83.1	90.3	95.8	95.3	93.8	90.2	92.3
	+ LSI	93.6	90.1	89.8	89.0	95.1	83.7	90.0	88.2	91.9	90.4	93.9	91.8	91.3	91.3	90.7
	+ ReCoVeR	**100.0**	**99.3**	**99.0**	98.3	**99.7**	90.0	98.3	**99.0**	99.6	**97.0**	**99.7**	96.9	98.6	**95.3**	97.9
	+ ReCoVeR+	99.7	98.9	**99.0**	**99.0**	99.6	**92.1**	**99.3**	97.9	**100.0**	96.3	99.6	**97.8**	**100.0**	95.2	**98.2**
	Gemma 2	70.4	80.9	84.9	82.3	66.4	64.5	81.7	77.6	70.5	82.3	79.3	79.5	71.1	68.8	76.1
	+ LSI	74.3	88.6	93.2	92.4	80.9	61.7	82.7	72.0	76.8	81.0	75.4	79.8	83.3	73.3	79.7
	+ ReCoVeR	96.2	98.6	96.4	94.0	98.6	77.0	95.5	92.9	97.1	**96.2**	**100.0**	95.5	**98.0**	90.9	94.8
	+ ReCoVeR+	**98.5**	**99.3**	**98.6**	**96.3**	**100.0**	**91.7**	**98.6**	**97.2**	**99.6**	93.5	99.2	**98.9**	97.5	**92.9**	**97.3**

Table 3: Cross-LC results on the LCB. For our learned steering function (ReCoVeR+), languages not seen during training are highlighted in (darker) grey.

	de		en		eu		fa		fr		sw		tr		zh		AVG	
Model	LPR	Acc	LPR	Acc	LPR	Acc	LPR	Acc	LPR	Acc	LPR	Acc	LPR	Acc	LPR	Acc	LPR	Acc
LLama 3.1	36.7	22.1	11.9	0.9	48.1	16.2	34.9	13.7	32.7	23.7	38.6	22.2	36.7	17.1	25.1	16.9	33.1	16.6
+ LSI	19.9	25.3	61.4	16.3	26.1	7.6	34.1	4.7	23.7	24.7	47.2	9.1	34.2	8.3	20.7	14.3	33.4	13.8
+ ReCoVeR	96.8	34.9	**98.7**	34.9	92.1	13.7	**98.9**	22.7	**97.2**	27.9	**72.4**	13.1	**96.5**	21.3	**96.6**	34.9	**93.7**	25.4
+ ReCoVeR+	**97.1**	**51.1**	79.5	**53.1**	**94.0**	**23.9**	97.3	**37.4**	95.6	**50.0**	69.4	**25.8**	95.0	**35.6**	85.5	**47.6**	91.4	**40.6**
Qwen 2.5	54.9	33.7	**94.8**	**60.8**	48.0	8.6	55.9	19.4	53.5	40.0	44.1	7.1	54.2	17.1	58.8	35.9	58.0	27.8
+ LSI	47.6	27.4	92.3	55.5	42.7	5.1	54.7	12.3	55.3	31.1	39.6	5.1	46.8	15.7	48.6	38.1	53.4	23.8
+ ReCoVeR	94.6	34.2	93.2	57.9	81.1	**13.2**	97.2	**14.7**	94.5	42.6	60.0	**13.6**	92.7	**28.7**	91.4	**51.9**	88.1	32.1
+ ReCoVeR+	**97.3**	**44.2**	93.2	57.9	**83.3**	12.7	**98.2**	11.4	**95.5**	**50.0**	**72.6**	9.6	**95.4**	27.3	**91.5**	45.5	**90.9**	**32.3**
Gemma 2	28.1	12.6	**98.1**	48.8	19.3	4.6	18.2	3.8	32.6	18.9	10.6	13.6	44.6	19.9	34.0	20.6	35.7	17.9
+ LSI	64.9	18.4	94.2	36.8	14.0	1.5	4.8	0.0	61.2	22.1	21.2	6.1	47.3	14.8	20.0	14.3	41.0	14.3
+ ReCoVeR	83.8	23.2	98.0	48.3	53.5	**15.2**	85.9	10.9	78.6	32.1	43.9	**18.7**	**94.3**	**35.6**	71.0	35.4	76.1	27.4
+ ReCoVeR+	**96.2**	**42.1**	98.0	**53.1**	**58.5**	7.6	**90.2**	**14.2**	**92.1**	**39.5**	**55.8**	16.7	93.8	24.5	**79.1**	**37.6**	**83.0**	**29.4**

Table 4: Cross-LC (LPR) and QA results (Acc) on MultiQ. Languages not seen in ReCoVeR+ training are highlighted in (darker) grey.

box) model and the steered model remains within 0.4 pp. With ReCoVeR we do not sacrifice task performance to improve language confusion.

Crosslingual Language Confusion. Cross-LC results (LLM instructed to reply in a concrete language, different from the prompt language) on LCB, MultiQ, and CrossSum are shown in Table 3, Table 13, Table 4, and Figure 3 (per language-pair results in Table 10), respectively. Owing to (1) computation of language vectors that is agnostic to any reference language (i.e., English) and (2) computation of the steering vectors using both representation r_{source} of the prompt language and r_{target} of the requested answer language, both ReCoVeR variants excel in Cross-LC: they massively outperform LSI across the board, for all three benchmarks, in zero-shot and few-shot evaluation and all three LLMs. ReCoVeR variants again outperform LSI most prominently for Gemma, but the gains are often large for the other two LLMs too: e.g., on CrossSum, ReCoVeR (+) yields +28pp over LSI (+24pp over the original model) for Qwen and +50pp (+46pp) for Llama. We also report the performance of language-specific neurons on LCB. Across all three models ReCoVeR (+) outperforms language-specific neurons. ReCoVeR consistently maintains or improves WPR. The additional training of ReCoVeR+ tends to improve the WPR over ReCoVeR.

We note that all three models exhibit dramatically worse Cross-LC performance on MultiQ (Table 4) than on the LCB dataset (Table 3), e.g., 91% vs. 33% for Llama. This is because the prompt language in MultiQ varies and is not fixed to English as in LCB. This shows that multilingual LLMs fail to comply with the requested response language much more often if the prompt is not in English. LSI actually consistently exhibits more Cross-LC than the corresponding base models on MultiQ and CrossSum (Figure 3) and this is also due to

the non-English prompts in those datasets. LSI computes its language vectors aiming to mitigate answering in the dominant language. However, for non-English prompts, the model is prone to answer in the prompt language rather than English, which seems to render LSI inutile for cross-lingual applications with non-English prompts.

Figure 1 shows example generations of Llama without and with ReCoVeR. Out-of-the-box Llama struggles to follow the instruction and reply in the correct language. Instead it replies in the source language. In the second example the content of the answer changes as well. ReCoVeR provides the correct answer but the sentence, while understandable, is not in correct grammar.

In contrast to Mono-LC results, in Cross-LC the unsupervised steering based on simple subtraction of language vectors (ReCoVeR) almost matches the performance of the learned steering (ReCoVeR+): this suggests that the difference $r_{source} - r_{target}$ already represents a very good steering vector for Cross-LC and that the learned steering does not capture much more than this difference either.

Generalization to Unseen Languages. On both LCB and MultiQ, we evaluate ReCoVeR+ on several target languages to which the model was not exposed during training of the steering intervention. The performance for those languages thus quantifies the extent of cross-lingual generalization of our learned steering function. The results in Table 3 (languages: it, ko, tr, vi) and Table 4 (languages: eu, sw, tr) indicate a successful cross-lingual of the mitigation of language confusion (i.e., ReCoVeR+ consistently matches or surpasses the performance of ReCoVeR). The same, however, does not consistently hold for task-specific performance on MultiQ: especially for Gemma, transfer of the learned steering function of ReCoVeR+ to an unseen language can lead to a substantial performance drop (e.g., -8pp for Basque or -11pp for Turkish) compared to the unsupervised steering (ReCoVeR). Although it is worth noting that even in these cases ReCoVeR+ almost always yields higher QA accuracy than the base model and steering with LSI.

5.2 Further Analyses

Leave-Out Layers. Previous work suggested that some layers produce representations that encode more language information than others (Bhattacharya and Bojar, 2023; Tang et al., 2024a). We thus next measure language confusion while omitting language steering in one transformer layer at a time. We run the experiment on Qwen, using the complex questions from LCB (languages: de, es, hi, ja, pt, zh). Figure 4 summarizes the results: we achieve best performance (90.5% LPR) when applying steering in *all* layers. We observe the largest performance drops if we remove the steering in layers 1 and 2: effective mitigation of language confusion seems to require very "early" steering. We again see larger drops if we remove steering from higher layers (L22-L27). Our finding that language-specific steering is more important at the bottom and top layers than in middle layers is in line with the hypothesis that the intermediate layers are responsible for reasoning, and thus rely on the English-centric representations (Zhao et al., 2024).

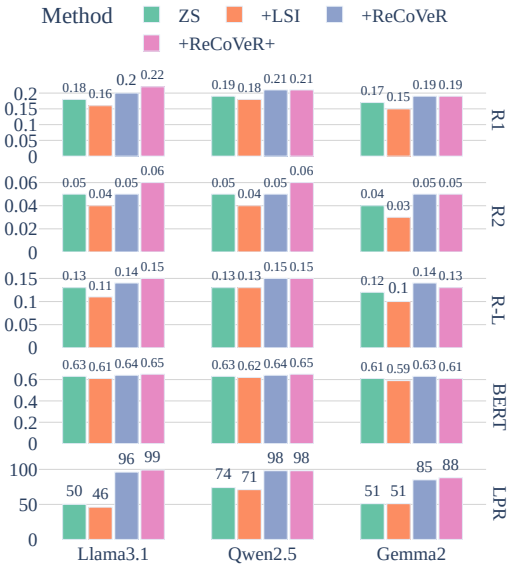


Figure 3: Cross-LC (metric: LPR) and summarization performance (Rouge-1: R1; Rouge-2: R2; Rouge-L: R-L; and BERT-Score: BERT) on the CrossSum dataset for the language pairs: en-es, es-fr, fr-tr, and tr-sw.

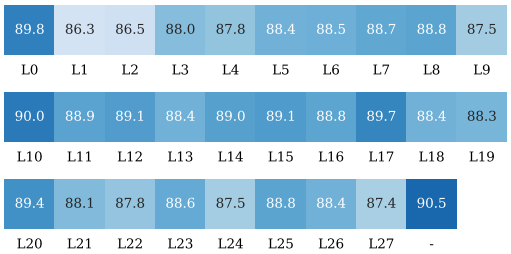


Figure 4: LPR for Qwen2.5 when we remove steering from individual layers.

Steering as the Only Language Indication. Prompts in the Cross-LC evaluation explicitly specify the expected response language: ReCoVeR steering then mitigates language confusion and helps LLMs generate text in the specified language.

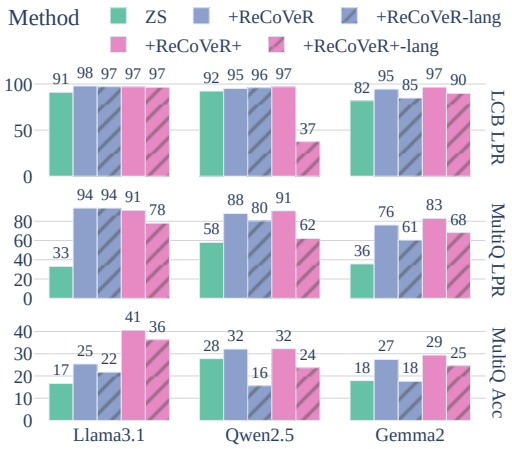


Figure 5: Cross-LC performance on LCB and MultiQ without language specification in the prompt.

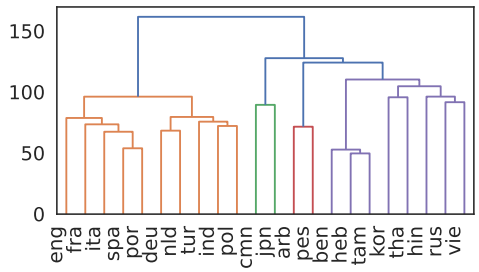


Figure 6: Visualization of the language representations of the last layer of Llama 3.1 using Agglomerative Hierarchical Clustering.

We next investigate to which extent ReCoVeR steering alone conditions the generation language, i.e., when we remove the answer language specification from the prompt. ReCoVeR thus has to steer the model away from the implicitly specified prompt language and towards the target language: this requires stronger steering compared to prompts with explicit language specifications, and we find that optimal $\alpha = 2$ for Qwen and $\alpha = 4$ for Gemma. For Llama, however, $\alpha = 1$ remains. The results, shown in Figure 5, reveal that, for Llama and Gemma, steering via ReCoVeR (+) alone is more effective than steering by only specifying the response language in the prompt (ZS).

Language Vectors. We next qualitatively analyze language vectors r_l by grouping them via hierarchical agglomerative clustering, to see if they reflect known relations between languages (e.g., typology/genealogy, shared script and/or shared vocabulary). Figure 6 displays the resulting dendrogram for Llama (Figure 7 shows Qwen and Gemma). Our language steers capture, to some extent, language families: all models generally group together Indo-European Languages and in particular the four Romance languages (French, Spanish, Italian and Portuguese; with Spanish and Portuguese having consistently most similar vectors). There is some evidence that a shared script drives the proximity: (1) representations from all three models tend to group Latin-script languages; and (2) Llama vectors put Chinese and Japanese close together: although Japanese is in a different language family and also typologically dissimilar to Chinese, it borrows Chinese scripts and a non-negligible portion of Chinese vocabulary. Qwen, on the other hand, represents Chinese similar to Hindi, Bengali, and Tamil. These languages are from different language families, and generally have little in common other than geographic proximity.

Negative Result: Position-Specific Steering. In ReCoVeR, we compute language-specific vectors at the level of layers, i.e., we steer representations of *all* tokens in a layer with the same steering vector. Initially, however, we intended to compute/learn position-specific steering vectors, i.e., a different steering vector for each layer and each token position. However, positional information largely gets lost due to averaging across samples: because of varying sentence structures and sequence lengths, different positions end up with very similar representations. Because of this, position-specific steering consistently produced worse results than layer-level steering (we report the results in §C).

6 Conclusion

We introduced ReCoVeR, a novel lightweight language steering approach. We first isolate language vectors using a multi-parallel corpus and then leverage those vectors for effective language steering of LLMs via (i) unsupervised steering arithmetic as well as (ii) learnable steering intervention, trained in a sample-efficient manner. Our extensive evaluation, encompassing three benchmarks and 18 languages, shows that, in contrast to prior approaches, ReCoVeR effectively mitigates language confusion without jeopardizing task performance. Future work could leverage linguistic information to obtain more effective language steering, e.g., vectors conditioned by (i) syntactic roles, exploiting multilingual dependency parsers (De Marneffe et al., 2021) or (ii) typological features, e.g., URIEL (Littell et al., 2017), to facilitate cross-lingual generalization (i.e., transfer) for learning steering vectors.

Limitations

Our experiments focus on a broad range of languages across different families, offering wide-ranging initial insights. While this provides substantial coverage, the applicability of the current results to the entirety of global languages requires further investigation.

We evaluate ReCoVeR on a diverse set of state-of-the-art multilingual model families, including Llama, Qwen, and Gemma. These represent widely adopted multilingual models, providing a strong basis for evaluating the generalizability of ReCoVeR. The results suggest that ReCoVeR is likely to generalize well beyond the models tested.

We use FLORES-200 as the multi-parallel data source to compute language representations. Although it provides meaningful language representations, demonstrating that our findings extend to other multi-parallel datasets requires further experiments. Moreover, we use all available samples in FLORES-200, and thus do not address the question of how many samples are minimally required to obtain meaningful language representations.

Acknowledgements

Hannah Sterz thanks the Cambridge Trust for their support via the International Scholarship. This work has been supported by a Royal Society University Research Fellowship *'Inclusive and Sustainable Language Technology for a Truly Multilingual World'* (no 221137) awarded to Ivan Vulić.

References

Guillaume Alain and Yoshua Bengio. 2017. Understanding intermediate layers using linear classifier probes. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*. OpenReview.net.

Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Jon Ander Campos, Yi Chern Tan, and 1 others. 2024. Aya 23: Open weight releases to further multilingual progress. *arXiv preprint arXiv:2405.15032*.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, Yuan-Fang Li, Yong-Bin Kang, and Rifat Shahriyar. 2023. CrossSum: Beyond English-centric cross-lingual summarization for 1,500+ language pairs. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2541–2564, Toronto, Canada. Association for Computational Linguistics.

Sunit Bhattacharya and Ondřej Bojar. 2023. Unveiling multilinguality in transformer models: Exploring language specificity in feed-forward networks. In *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 120–126, Singapore. Association for Computational Linguistics.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, and 1 others. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2.

Yuanpu Cao, Tianrong Zhang, Bochuan Cao, Ziyi Yin, Lu Lin, Fenglong Ma, and Jinghui Chen. 2024. Personalized steering of large language models: Versatile steering vectors through bi-directional preference optimization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Marta R Costa-Jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, and 1 others. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Marie-Catherine De Marneffe, Christopher D Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal dependencies. *Computational linguistics*, 47(2):255–308.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Evan Hernandez, Belinda Z. Li, and Jacob Andreas. 2024. Inspecting and editing knowledge representations in language models. In *First Conference on Language Modeling*.

Carolin Holtermann, Paul Röttger, Timm Dill, and Anne Lauscher. 2024. Evaluating the elementary multilingual capabilities of large language models with MultiQ. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4476–4494, Bangkok, Thailand. Association for Computational Linguistics.

Ole Jorgensen, Dylan Cope, Nandi Schoots, and Murray Shanahan. 2024. Improving activation steering in language models with mean-centring. In *Responsible Language Models Workshop at AAAI-24*.

Takeshi Kojima, Itsuki Okimura, Yusuke Iwasawa, Hitomi Yanaka, and Yutaka Matsuo. 2024. On the multilingual ability of decoder-based pre-trained language models: Finding and controlling language-specific neurons. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6919–6971, Mexico City, Mexico. Association for Computational Linguistics.

Faisal Ladhak, Esin Durmus, Claire Cardie, and Kathleen McKeown. 2020. WikiLingua: A new benchmark dataset for cross-lingual abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4034–4048, Online. Association for Computational Linguistics.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, and 1 others. 2024. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.

Patrick Littell, David R Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. Uriel and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 8–14.

Sheng Liu, Haotian Ye, Lei Xing, and James Y. Zou. 2024. In-context vectors: Making in context learning more effective and controllable through latent space steering. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Kelly Marchisio, Wei-Yin Ko, Alexandre Berard, Théo Dehaze, and Sebastian Ruder. 2024. Understanding and mitigating language confusion in LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6653–6677, Miami, Florida, USA. Association for Computational Linguistics.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2024. The linear representation hypothesis and the geometry of large language models. In *International Conference on Machine Learning*, pages 39643–39666. PMLR.

Ricardo Rei, Marcos Treviso, Nuno M. Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte Alves, Luisa Coheur, Alon Lavie, and André F. T. Martins. 2022. CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 634–645, Abu Dhabi, United Arab Emirates (Hybrid). 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.

Shashwat Singh, Shauli Ravfogel, Jonathan Herzig, Roee Aharoni, Ryan Cotterell, and Ponnurangam Kumaraguru. 2024. Representation surgery: Theory and practice of affine steering. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

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.

Alessandro Stolfo, Vidhisha Balachandran, Safoora Yousefi, Eric Horvitz, and Besmira Nushi. 2025. Improving instruction-following in language models through activation steering. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Nishant Subramani, Nivedita Suresh, and Matthew Peters. 2022. Extracting latent steering vectors from pretrained language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 566–581, Dublin, Ireland. Association for Computational Linguistics.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Wayne Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024a. 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.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024b. 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.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, and 1 others. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2024.

Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. 2023. Activation addition: Steering language models without optimization. *arXiv e-prints*, pages arXiv–2308.

Weixuan Wang, JINGYUAN YANG, and Wei Peng. 2025. Semantics-adaptive activation intervention for LLMs via dynamic steering vectors. In *The Thirteenth International Conference on Learning Representations*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Xie Yunfan, Lixin Zou, Dan Luo, Min Tang, Chenliang Li, Xiangyang Luo, and Liming Dong. 2025. Mitigating language confusion through inference-time intervention. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8418–8431, Abu Dhabi, UAE. Association for Computational Linguistics.

Jason Zhang and Scott W Viteri. 2025. Uncovering latent chain of thought vectors in large language models. In *Workshop on Neural Network Weights as a New Data Modality*.

Yiran Zhao, Wenxuan Zhang, Guizhen Chen, Kenji Kawaguchi, and Lidong Bing. 2024. How do large language models handle multilingualism? In *Advances in Neural Information Processing Systems*, volume 37, pages 15296–15319. Curran Associates, Inc.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, and 1 others. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

A Trainings and Evaluation Details

A.1 Hyperparameters

ReCoVeR depends on the choice of hyperparameters. Choosing suitable α and whether to restore the norm is crucial for reducing language confusion. We choose the hyperparameters based on the performance on a smaller dataset of 600 machine translated samples from the alpaca dataset. We measure LPR and exclude hyperparameters that result in unreadable text; e.g. by repeating a word or sequence of words corresponding to the language. The hyperparameters are listed in Table 5. The monolingual scenario requires less steering and a smaller α, except for Qwen.

Model	Version	Task	alpha	beta	norm
Llama 3.1	ReCoVeR	cross	0.2	-	true
	ReCoVeR	mono	0.05	-	true
	ReCoVeR+	both	0.1	0.9	true
Qwen 2.5	ReCoVeR	cross	1	-	true
	ReCoVeR	mono	2	-	true
	ReCoVeR+	both	1	0.9	true
Gemma 2	ReCoVeR	cross	2	-	false
	ReCoVeR	mono	0.5	-	true
	ReCoVeR+	both	2	0.9	true

Table 5: Hyperparameter used in the experiments, covering alpha, beta, and whether to restore the norm.

LSI requires choosing hyperparameters τ, γ. We list the hyperparameters in 6. We base the search space on the range used in the original paper $\tau \in [0.02, 0.04, 0.06, 0.08, 0.1]$ and $\gamma \in [0.2, 0.4, 0.6, 0.8, 1.0]$. However, for Gemma 2 we observe a model collapse for parameters in this range. Therefore, we reduce γ to 0.01, generating coherent text in the target language.

Model	Task	τ	γ
Llama 3.1	cross	0.06	0.6
	mono	0.04	0.6
Qwen 2.5	cross	0.06	0.2
	mono	0.04	0.4
Gemma 2	cross	0.04	0.01
	mono	0.04	0.01

Table 6: Hyperparameter used for LSI in the experiments

A.2 Learnable Steering Function: Training Details

We train ReCoVeR+ with the following hyperparameters: we train for 1 epoch with a learning rate of $1e-6$ and Adam optimiser and 0.2 dropout, and a rank of 32. The model-specific hyperparameters are listed in Table 5.

To isolate the influence of the language vectors on the generated text, the input prompts are formulated without explicit information specifying the target language. This ensures that the language selection in the output is guided by the learned steering vectors, rather than by direct cues within the prompts themselves.

A.3 Dataset Details

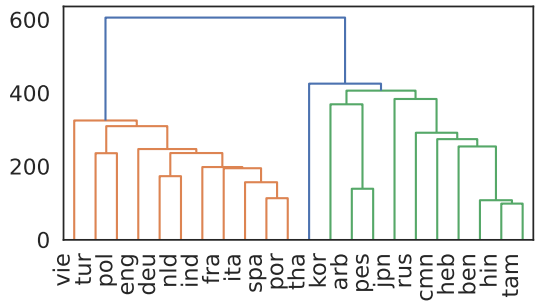
We create a multilingual instruction tuning dataset that contains monolingual (prompt and answer are in the same language) and crosslingual (prompt and

answer are in different languages) samples. Table 7 shows the number of samples per language for both. The dataset covers 18 languages from diverse language families and with varying scripts.

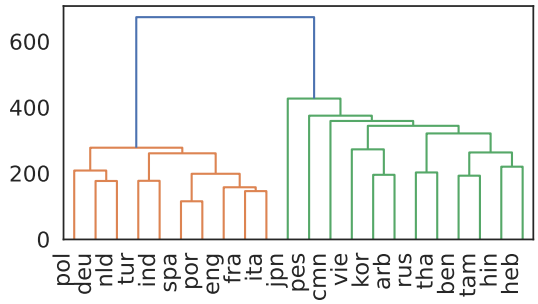
Recognizing that state-of-the-art LLMs are more prone to language confusion in crosslingual versus monolingual generation, we have increased the proportion of crosslingual samples compared to monolingual samples for each language in our training data.

To evaluate the translation qualities, we compute the Comet-Kiwi(Rei et al., 2022) scores for the questions and answers. We group the scores into brackets and visualize them in Figure 8. The majority of the questions get scores > 0.8. For the translated answers, the scores are more in the 0.6 - 0.8 bin, probably due to the long sequence length, which makes it harder to determine whether two sequences are translations of each other.

B Language Representations



(a) Qwen 2.5



(b) Gemma 2

Figure 7: Visualization of the language representations of the last layer of Qwen 2.5 and Gemma 2 using Agglomerative Hierarchical Clustering.

C Position-Specific Steering

We explore including positions in the computation of the steering vectors. Instead of averaging across positions, the average hidden state is computed per position. This increases the size of the language vector by k times, where k is the number of positions we consider. However, results in Table 8 show that there is no advantage to having this information. In fact, it tends to harm performance.

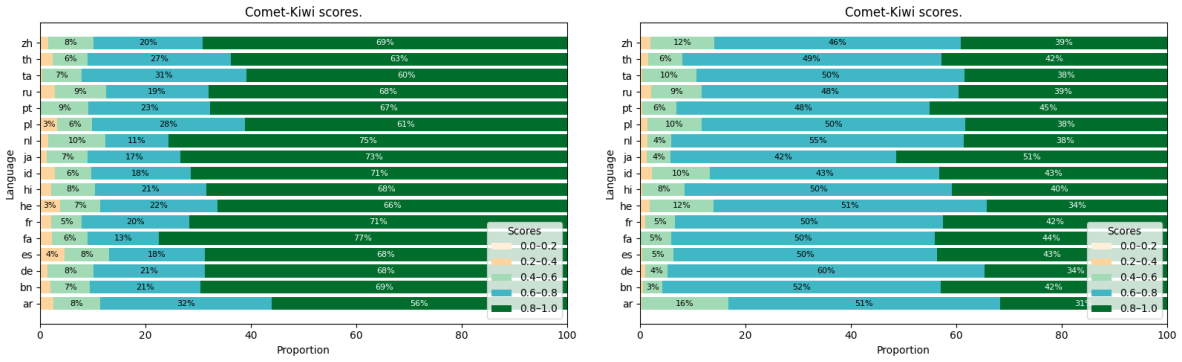
D Detailed Results

In the main part of the paper we cover monolingual scenarios and crosslingual summarization only aggregated across all languages. Table 11 and Table 12 show the monolingual performance for the individual languages. Across both datasets, the LPR consistently surpasses 90% for most evaluations, indicating near-optimal performance.

The crosslingual summarization scores on CrossSum are shown in Table 10. The Cross-LC LPR is shown in Table 9. They illustrate that ReCoVeR and ReCoVeR+ can improve Cross-LC while maintaining or improving the summarization performance.

Lang	Family	Script	Src	Tgt	Mono	Total
ar	Afro-Asiatic	Arabic	256	230	51	537
bn	Indo-European	Bengali	230	263	65	558
de	Indo-European	Latin	163	244	45	452
en	Indo-European	Latin	207	158	30	395
es	Indo-European	Latin	208	177	29	414
fa	Indo-European	Perso-Arabic	235	221	31	487
fr	Indo-European	Latin	213	255	63	531
he	Afro-Asiatic	Hebrew	172	119	39	330
hi	Indo-European	Devanagari	206	186	25	417
id	Austronesian	Latin	182	222	35	439
ja	Austronesian	Han, Hiragana, Katakana	308	242	67	617
nl	Indo-European	Devanagari	145	178	40	363
pl	Indo-European	Latin	193	157	49	399
pt	Indo-European	Latin	156	221	39	416
ru	Indo-European	Cyrillic	182	153	34	369
ta	Dravidian	Tamil	178	195	39	412
th	Kra-Dai	Thai	250	213	37	500
zh	Sino-Tibetan	Han	153	203	45	401
Total			3637	3637	763	4400

Table 7: The number of samples that have each language as a source or target language in for the crosslingual samples and the number of monolingual samples.



(a) Comet-Kiwi Scores on the translated Questions.

(b) Comet-Kiwi Scores on the translated Answer.

Figure 8: Translation Quality based on Comet-Kiwi scores.

Model	Method	Cross	Mono
LLaMA	+ Ours	97.9	99.1
	+ Ours+Pos	96.7	98.6
	Δ	-1.2	-0.5
Qwen	+ Ours	95.2	97.7
	+ Ours+Pos	95.8	93.3
	Δ	+0.6	-4.4
Gemma	+ Ours	94.5	87.8
	+ Ours+Pos	89.9	89.5
	Δ	-4.6	+1.7

Table 8: LPR on the LCB with language representations for each position.

Model	es	fr	tr	sw	AVG
Llama	100.0	0.0	96.4	3.0	$49.8_{\pm 0.004}$
+ LSI	100.0	10.2	71.1	3.0	$46.1_{\pm 0.003}$
+ ReCoVeR	100.0	100.0	98.8	85.2	$96.0_{\pm 0.002}$
+ ReCoVeR+	100.0	100.0	100.0	95.6	$98.9_{\pm 0.003}$
Qwen	98.3	91.8	95.1	9.6	$73.7_{\pm 0.007}$
+ LSI	97.4	75.5	92.7	20.0	$71.4_{\pm 0.009}$
+ ReCoVeR	98.8	100.0	98.8	94.8	$98.1_{\pm 0.006}$
+ ReCoVeR+	100.0	100.0	100.0	91.9	$98.0_{\pm 0.002}$
Gemma 2	84.2	89.8	30.1	0.0	$51.0_{\pm 0.009}$
+ LSI	98.7	99.0	4.8	0.0	$50.6_{\pm 0.003}$
+ ReCoVeR	100.0	100.0	98.8	41.0	$85.0_{\pm 0.005}$
+ ReCoVeR+	100.0	100.0	100.0	52.6	$88.1_{\pm 0.004}$

Table 9: Cross-LC results on CrossSum for the language pairs: $en \rightarrow es$, $es \rightarrow fr$, $fr \rightarrow tr$, $tr \rightarrow sw$, denoted by the target language, and the standard error for the average across languages.

	$en \to es$				$es \to fr$				$fr \to tr$				$tr \to sw$				AVG			
Model	R-1	R-2	R-L	Bert	R-1	R-2	R-L	Bert	R-1	R-2	R-L	Bert	R-1	R-2	R-L	Bert	R-1	R-2	R-L	Bert
LLama 3.1	0.29	0.08	0.20	0.72	0.17	0.03	0.13	0.69	0.20	0.07	0.14	0.53	0.04	0.01	0.04	0.60	$0.18_{\pm 0.002}$	$0.05_{\pm 0.001}$	$0.13_{\pm 0.002}$	$0.63_{\pm 0.002}$
+ LSI	0.28	0.07	0.19	0.71	0.15	0.04	0.11	0.67	0.16	0.04	0.11	0.45	0.04	0.01	0.04	0.60	$0.16_{\pm 0.003}$	$0.04_{\pm 0.001}$	$0.11_{\pm 0.002}$	$0.61_{\pm 0.002}$
+ ReCoVeR	0.29	0.08	0.20	0.71	0.23	0.07	0.16	0.69	0.19	0.05	0.15	0.52	0.08	0.01	0.07	0.61	$0.20_{\pm 0.003}$	$0.05_{\pm 0.002}$	$0.14_{\pm 0.002}$	$0.63_{\pm 0.002}$
+ ReCoVeR+	0.30	0.09	0.21	0.72	0.22	0.07	0.15	0.69	0.20	0.06	0.15	0.54	0.14	0.02	0.10	0.66	$0.22_{\pm 0.003}$	$0.06_{\pm 0.002}$	$0.15_{\pm 0.002}$	$0.65_{\pm 0.002}$
Qwen 2.5	0.29	0.07	0.19	0.71	0.24	0.07	0.16	0.70	0.19	0.05	0.14	0.49	0.05	0.01	0.04	0.61	$0.19_{\pm 0.003}$	$0.05_{\pm 0.002}$	$0.13_{\pm 0.002}$	$0.63_{\pm 0.002}$
+ LSI	0.28	0.07	0.18	0.71	0.21	0.04	0.14	0.69	0.19	0.04	0.13	0.48	0.06	0.01	0.05	0.61	$0.18_{\pm 0.002}$	$0.04_{\pm 0.001}$	$0.13_{\pm 0.002}$	$0.62_{\pm 0.001}$
+ ReCoVeR	0.29	0.08	0.19	0.71	0.25	0.07	0.17	0.71	0.18	0.05	0.13	0.49	0.13	0.01	0.10	0.67	$0.21_{\pm 0.003}$	$0.05_{\pm 0.002}$	$0.15_{\pm 0.002}$	$0.65_{\pm 0.002}$
+ ReCoVeR+	0.30	0.09	0.20	0.72	0.25	0.08	0.18	0.72	0.20	0.05	0.15	0.52	0.11	0.01	0.08	0.64	$0.21_{\pm 0.003}$	$0.06_{\pm 0.002}$	$0.15_{\pm 0.002}$	$0.65_{\pm 0.002}$
Gemma 2	0.26	0.07	0.18	0.71	0.23	0.06	0.17	0.71	0.12	0.03	0.10	0.41	0.05	0.01	0.05	0.61	$0.17_{\pm 0.003}$	$0.04_{\pm 0.002}$	$0.12_{\pm 0.002}$	$0.61_{\pm 0.002}$
+ LSI	0.27	0.07	0.19	0.71	0.19	0.04	0.13	0.69	0.07	0.01	0.06	0.35	0.04	0.01	0.04	0.61	$0.15_{\pm 0.002}$	$0.03_{\pm 0.001}$	$0.10_{\pm 0.002}$	$0.59_{\pm 0.002}$
+ ReCoVeR	0.29	0.08	0.20	0.71	0.25	0.07	0.18	0.71	0.19	0.05	0.14	0.52	0.05	0.01	0.04	0.57	$0.20_{\pm 0.003}$	$0.05_{\pm 0.002}$	$0.14_{\pm 0.002}$	$0.63_{\pm 0.002}$
+ ReCoVeR+	0.29	0.08	0.19	0.71	0.26	0.07	0.18	0.71	0.16	0.05	0.12	0.50	0.05	0.01	0.04	0.53	$0.19_{\pm 0.003}$	$0.05_{\pm 0.002}$	$0.13_{\pm 0.002}$	$0.61_{\pm 0.002}$

Table 10: The crosslingual summarization performance in Rouge-1, Rouge-2, Rouge-L, and Bert Score on CrossSum with the standard error for the average across languages.

Model	ar	de	en	es	fr	hi	id	it	ja	ko	pt	ru	tr	vi	zh	avg
LLama 3.1	99.7	100.0	98.5	99.0	100.0	100.0	93.0	100.0	99.0	100.0	95.5	100.0	99.0	100.0	96.5	98.7
+ LSI	100.0	100.0	99.0	99.7	99.7	100.0	93.0	100.0	99.0	100.0	99.0	100.0	99.0	99.0	98.0	99.0
+ Lang Neuron	100.0	100.0	100.0	99.7	100.0	99.0	95.0	100.0	99.0	100.0	97.5	100.0	100.0	100.0	98.0	99.2
+ ReCoVeR	99.3	100.0	99.5	99.7	99.3	99.0	97.0	99.0	100.0	99.0	98.5	100.0	97.0	100.0	99.0	**99.1**
+ ReCoVeR+	99.7	100.0	99.5	98.7	100.0	100.0	94.0	99.0	100.0	100.0	96.5	100.0	100.0	99.0	99.5	**99.1**
Qwen 2.5	98.0	99.0	100.0	98.7	98.7	100.0	98.0	100.0	92.0	97.9	96.0	100.0	98.0	100.0	98.0	98.3
+ LSI	98.3	98.0	100.0	99.3	99.0	100.0	93.0	100.0	96.0	96.8	95.5	99.0	97.0	100.0	97.5	98.0
+ Lang Neuron	96.7	97.0	100.0	98.7	99.0	91.0	94.0	100.0	89.0	100.0	94.5	94.8	90.9	99.0	98.5	96.2
+ ReCoVeR	98.7	99.0	100.0	99.3	98.3	99.0	96.0	100.0	91.0	99.0	95.5	95.0	97.0	100.0	97.5	97.7
+ ReCoVeR+	99.3	98.0	100.0	99.7	99.7	98.0	96.0	98.0	96.0	100.0	96.5	100.0	99.0	99.0	99.0	**98.5**
Gemma 2	83.2	98.0	98.5	95.7	94.3	67.0	71.0	94.0	99.0	99.0	97.0	93.0	99.0	80.0	57.5	88.4
+ LSI	92.7	96.0	98.5	97.7	97.3	87.0	68.0	97.0	91.0	97.0	92.0	95.0	94.0	84.0	66.0	90.2
+ Lang Neuron	85.2	93.9	99.5	97.7	93.3	81.0	58.0	97.0	96.0	100.0	96.5	93.0	99.0	85.0	70.0	89.7
+ ReCoVeR	82.9	97.0	98.5	95.3	93.0	66.0	70.0	95.0	95.0	100.0	96.0	93.0	100.0	78.0	56.5	87.8
+ ReCoVeR+	98.0	100.0	100.0	99.0	99.3	100.0	90.0	100.0	98.0	100.0	96.5	99.0	100.0	99.0	93.0	**98.1**

Table 11: Mono-LC results on the LCB. For our learned steering function (ReCoVeR+), languages not seen during training are highlighted in (darker) grey.

	de		en		fr		tr		zh		AVG	
Model	LPR	Acc	LPR	Acc	LPR	Acc	LPR	Acc	LPR	Acc	LPR	Acc
LLama 3.1	96.4	65.5	94.3	67.0	95.4	65.0	98.9	57.0	87.5	67.5	94.5	**64.4**
LSI	97.4	55.0	93.8	59.5	98.0	53.5	99.4	42.0	90.0	54.0	95.7	52.8
+ ReCoVeR	94.4	64.0	93.9	60.5	93.9	61.0	96.1	52.5	90.9	68.5	93.8	61.3
+ ReCoVeR+	97.8	68.0	96.3	70.0	97.9	67.0	95.8	47.0	91.0	58.5	**95.8**	62.1
Qwen 2.5	90.5	67.5	92.5	59.5	90.5	58.0	89.5	35.0	90.5	89.0	90.7	61.8
LSI	93.0	43.5	93.5	59.0	93.5	52.5	92.2	22.5	91.5	77.5	92.7	51.0
+ ReCoVeR	90.0	60.0	95.5	68.5	92.5	67.0	91.2	38.0	91.0	80.0	92.0	62.7
+ ReCoVeR+	94.3	69.0	93.4	72.5	94.2	70.5	94.9	40.5	89.4	80.0	**93.2**	**66.5**
Gemma 2	91.4	29.5	93.5	30.0	92.5	42.0	96.7	42.5	84.9	48.5	91.8	38.5
LSI	94.5	28.0	95.5	36.5	96.5	36.5	96.8	32.5	84.5	40.5	**93.5**	34.8
+ ReCoVeR	91.9	26.0	93.5	31.5	93.0	44.0	96.7	43.0	82.9	47.0	91.6	38.3
+ ReCoVeR+	92.4	45.0	94.5	47.0	93.9	50.0	92.9	42.0	90.8	53.0	92.9	**47.4**

Table 12: The LPR and Accuracy on the monolingual setup of the MultiQ.

	Monolingual							Crosslingual						
	ar	hi	ja	ko	ru	zh	avg	ar	hi	ja	ko	ru	zh	avg
LLama 3.1	98.3	100.0	100.0	99.0	96.0	98.5	98.4	94.0	97.2	93.4	93.5	93.4	96.9	94.7
LSI	99.3	100.0	100.0	98.0	97.0	96.9	98.5	91.9	99.2	94.8	97.1	96.0	97.3	96.0
ReCoVeR	98.0	100.0	98.0	96.0	97.0	97.0	97.7	97.9	98.3	93.8	95.4	98.6	96.6	96.7
ReCoVeR+	99.7	100.0	100.0	100.0	100.0	99.5	**99.9**	94.7	98.3	95.9	96.5	98.6	97.5	**96.9**
Qwen 2.5	100.0	99.0	98.9	100.0	98.0	99.0	99.2	94.3	96.7	89.4	95.5	93.5	96.2	94.3
LSI	100.0	100.0	99.0	98.9	100.0	100.0	**99.6**	94.3	98.1	89.9	95.7	94.4	94.8	94.5
ReCoVeR	99.7	100.0	98.9	100.0	95.7	99.5	99.0	96.2	98.3	91.7	95.3	94.8	88.0	94.0
ReCoVeR+	100.0	99.0	100.0	99.0	100.0	99.5	**99.6**	96.7	97.5	93.8	94.8	97.1	98.4	**96.4**
Gemma 2	96.7	100.0	91.9	100.0	93.6	96.7	96.5	86.2	98.8	98.7	90.9	93.5	98.6	**94.5**
LSI	96.6	98.9	100.0	97.9	96.8	96.1	97.7	69.0	91.4	88.0	86.6	88.4	97.6	86.8
ReCoVeR	95.6	98.5	94.7	100.0	94.6	96.5	96.7	91.0	97.2	83.6	94.4	94.2	91.8	92.0
ReCoVeR+	98.6	100.0	100.0	100.0	96.0	100.0	**99.1**	86.2	96.7	92.9	95.9	97.6	92.6	93.6

Table 13: WPR on the monolingual and crosslingual portion of the LCB.