LEVERAGING PRETRAINED KNOWLEDGE AT INFERENCE TIME: LORA-GATED CONTRASTIVE DECODING FOR MULTILINGUAL FACTUAL LANGUAGE GENERATION IN ADAPTED LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) adapted to specific languages through continual pretraining or instruction tuning often suffer from catastrophic forgetting, which can lead to factual inaccuracies. This issue is particularly pronounced in multilingual settings, where adaptation may override general world knowledge with language-specific patterns. We propose LoRA-Gated Contrastive Decoding (LGCD), a training-free inference-time decoding framework that improves factuality in language-adapted LLMs by leveraging knowledge from the original pretrained model. LGCD operates by (1) extracting factual representations from Feed-Forward Network (FFN) layers via LoRA-based decomposition, approximating pretrained knowledge, (2) dynamically gating decoding based on token-level confidence, and (3) applying contrastive decoding with Top-K masking to revise uncertain predictions by referencing the approximated representation of pretrained knowledge. LGCD requires no additional training or access to the original pretraining data. Extensive experiments with LGCD on multilingual multiple-choice and long-form QA tasks across nine languages demonstrate its strong effectiveness in mitigating hallucinations and enhancing factual accuracy in language-adapted models. These results further indicate that pretrained knowledge can be strategically reintroduced during decoding to promote factual multilingual generation.

1 INTRODUCTION

Large Language Models (LLMs) have demonstrated remarkable capabilities across a wide range of natural language tasks. A common practice to enhance their performance for specific languages or domains involves continual pretraining (CPT) or instruction fine-tuning (Gururangan et al., 2020; Zhang et al., 2024; Huang et al., 2023). While these adaptation techniques often inject new knowledge and improve task-specific abilities in the target language or domain, they frequently suffer from a critical drawback: *catastrophic forgetting* (Luo et al., 2023; OLMo et al., 2024; Li & Lee, 2024; Li et al., 2024; Kalajdzievski, 2024). This phenomenon leads to the degradation of general knowledge acquired during the initial pretraining phase, often resulting in increased factual inaccuracies or hallucinations (Ji et al., 2023; Luo et al., 2023; Li & Lee, 2024). Empirical studies confirm that LLMs undergoing CPT or instruction tuning can lose previously learned knowledge, sometimes prioritizing stylistic alignment or fluency in the target language over the factual consistency inherent in the original model (Luo et al., 2023).

Mitigating catastrophic forgetting during adaptation is challenging. Ideally, one would retrain the model using a mixture of the original pretraining data and the new adaptation data. However, the original pretraining datasets for many state-of-the-art LLMs (e.g., LLaMA, Qwen) are generally undisclosed and inaccessible, though efforts towards fully open models like OLMo exist (OLMo et al., 2024). Furthermore, retraining from scratch or even extensive CPT demands prohibitive computational resources and time. Although various techniques aim to reduce forgetting during the training process (Gu et al., 2024; Huang et al., 2024; He et al., 2024; Wang et al., 2023b; Vo et al., 2024), they remain limited in preserving general knowledge, especially when adapting to

new domains or languages. This limitation motivates the exploration of alternative approaches that can enhance the factuality of adapted LLMs without requiring further training or access to original pretraining data.

Recent research has highlighted the role of Feed-Forward Network (FFN) layers within the Transformer architecture as key-value memories, crucial for storing factual knowledge acquired during pretraining (Geva et al., 2020; Qiu et al., 2024; Dai et al., 2023). Inspired by this understanding, we hypothesize that the knowledge implicitly stored within the FFN weights of the original pretrained model can be explicitly leveraged to support the generation process of an adapted (e.g., continually pretrained or instruction-tuned) model at inference time, thereby improving its factual accuracy.

In this work, we propose LoRA-Gated Contrastive Decoding (LGCD), a novel training-free decoding method designed to enhance the factuality of LLMs, particularly those adapted for specific languages or domains. LGCD addresses the inherent trade-off between domain-specific fluency and general factual knowledge by dynamically switching between decoding strategies based on token-level confidence and applying contrastive decoding when necessary.

The framework of LGCD is characterized by three key components: First, it performs LoRA-based factual knowledge extraction from FFN layers and obtains a lightweight approximation of the pretrained model (PTM), by computing parameter differences between pretrained and adapted models and decomposing them using Singular Value Decomposition (SVD) to recover factual knowledge in FFN layers without modifying the language-adapted model (LAM). Second, it employs confidence-based dynamic gating that measures token-level confidence from the LAM and determines when to trigger factual knowledge injection, ensuring that domain fluency is preserved when the model is confident while leveraging pretrained knowledge when uncertainty arises. Third, it implements contrastive decoding with Top-K masking, which computes contrastive logits by subtracting the LAM's logits from the logits of the approximated PTM (aPTM), and applies this correction only to the top-K candidates predicted by the LAM. This selective adjustment injects factual knowledge while minimizing disruption to fluent generation.

We conduct a comprehensive evaluation of LGCD across nine diverse languages, highlighting its broad applicability in multilingual settings. Our experiments demonstrate LGCD's effectiveness across multiple evaluation settings, including multilingual multiple-choice benchmarks such as Global MMLU (Singh et al., 2024) and multilingual TruthfulQA (Dac Lai et al., 2023) for domain-specific and general factual knowledge, long-form generation benchmarks such as Multi-FAct (Shafayat et al., 2024) for factual consistency, and long-form medical QA tasks for precise knowledge grounding in high-stakes domains.

Our contributions are threefold:

1. We propose LGCD, a novel training-free, decoding-time framework to mitigate hallucination and enhance factuality in language-adapted LLMs by leveraging knowledge from the original pretrained model through dynamic model switching and contrastive decoding.
2. We introduce specific techniques within LGCD, including LoRA-based knowledge extraction from FFN layers, confidence-based dynamic gating for token-level decision making, and contrastive decoding with Top-K masking.
3. We provide extensive empirical evidence demonstrating LGCD's effectiveness across multilingual multiple-choice QA and long-form generation tasks, using nine languages and twelve models. Our approach consistently outperforms adapted models without requiring additional training or external resources.

2 Related work

2.1 Hallucination Mitigation in LLM

Addressing hallucinations in LLMs involves various strategies, including improvements in training data and model architecture, fact-checking mechanisms, and integrating external knowledge sources like retrieval systems or knowledge graphs (Izacard & Grave, 2020; Wang et al., 2023a). While effective, the aforementioned external methods often introduce complexity or dependencies. Our proposed method, LGCD, focuses on an internal, decoding-time approach to mitigate hallucination without requiring external models or significant architectural changes.

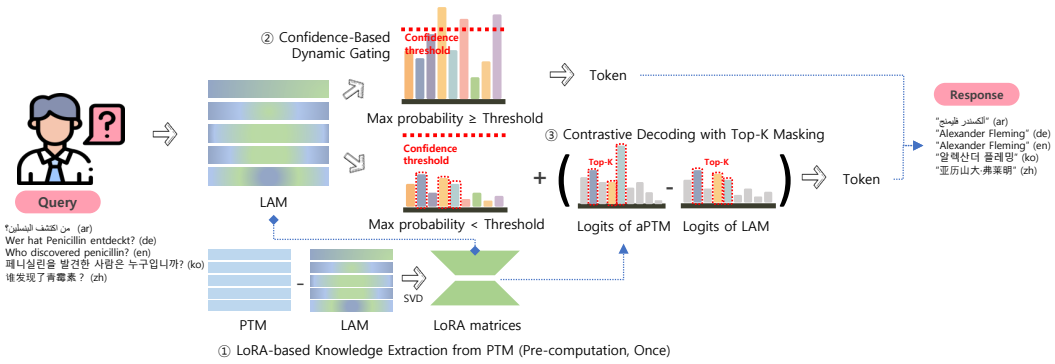

Figure 1: Overview of the LGCD framework.

2.2 Decoding Strategies for Factuality

Factual consistency in LLM decoding is often improved by tweaking output probabilities. For instance, Contrastive search (Su et al., 2022) combats repetition by picking tokens that are both likely and semantically distinct from prior context. Similarly, DoLa (Chuang et al., 2024) leverages deeper layers by contrasting logits across different internal layers of the same model. These methods reshape probability distributions based on disagreements or differences within the model or its variants. Our LGCD takes a different approach. Instead of merely contrasting probabilities, LGCD explicitly extracts and integrates factual knowledge from FFN layers of the original PTM–a process that is performed only once. This knowledge then directly influences the LAM's logits via a confidence-gated mechanism, offering a training-free solution to inject specific factual signals and enhance accuracy in adapted LLMs without further access to the PTM during inference.

2.3 Knowledge in Feed-Forward Networks

Previous research has shown that FFN layers within transformer models serve as key repositories for factual and world knowledge, often interpreted as key-value memories (Geva et al., 2020; Qiu et al., 2024; Dai et al., 2023). This perspective suggests that structured knowledge is encoded within their weights. Our LGCD leverages this by explicitly recovering factual knowledge from FFN layers of the original PTM. This is vital because catastrophic forgetting can degrade such knowledge in LAMs. LGCD achieves this through LoRA-based factual knowledge extraction: it calculates parameter differences between the PTM and LAM, then uses SVD to pinpoint and recover PTM's factual knowledge without altering the LAM. This precisely extracted knowledge is then dynamically injected during decoding, central to LGCD's factuality enhancement.

3 Methodology

This section introduces LGCD, a training-free decoding framework that combines the strengths of LAM and PTM. LGCD addresses the inherent trade-off between domain-specific fluency and general factual knowledge by dynamically switching between models based on token-level confidence and applying contrastive decoding when necessary. Figure 1 illustrates our framework.

The core motivation behind LGCD is grounded in the observation that FFN layers in Transformer architectures act as key-value neural memories for factual knowledge (Geva et al., 2020; Dai et al., 2023; Qiu et al., 2024). During CPT or instruction fine-tuning, this knowledge can be degraded due to catastrophic forgetting. LGCD mitigates this by explicitly recovering factual knowledge from the pretrained model's FFN layers and dynamically injecting it during decoding.

The LGCD framework consists of three components: (1) LoRA-based factual knowledge extraction from FFN layers, (2) confidence-based dynamic gating, and (3) contrastive decoding with Top-K masking.

3.1 LoRA-based Knowledge Extraction from FFN Layers

To capture factual knowledge preserved in the pretrained model, LGCD performs LoRA extraction from all FFN layers. Specifically, for each FFN layer ℓ, we compute the parameter difference between the pretrained model M_{PTM} and the language-adapted model M_{LAM}:

$$\Delta W_\ell = W_\ell^{\text{PTM}} - W_\ell^{\text{LAM}} \tag{1}$$

We then apply SVD:

$$\Delta W_\ell = U_\ell \Sigma_\ell V_\ell^\top \tag{2}$$

LoRA matrices are constructed by retaining the top-r singular components:

$$A_\ell = U_\ell[:, :r] \cdot \sqrt{\Sigma_\ell[:r]} \tag{3}$$

$$B_\ell = \sqrt{\Sigma_\ell[:r]} \cdot V_\ell^\top[:r, :] \tag{4}$$

The pretrained FFN weight is approximated as:

$$W_\ell^{\text{aPTM}} = W_\ell^{\text{LAM}} + A_\ell B_\ell \tag{5}$$

Notably, this process is performed only once in the entire framework and allows LGCD to retrieve factual knowledge from the PTM without modifying the LAM directly or incurring additional memory overhead from deploying separate models. It is applied only to the FFN layers, leaving all other components of the LAM unchanged. To empirically validate this design decision, we compare different layer-wise LoRA approximation strategies in Appendix A.3 and find that targeting only FFN layers yields the best performance.

3.2 Confidence-Based Dynamic Gating

At each decoding step t, LGCD first queries the LAM to compute logits $\mathbf{l}_t^{\text{LAM}}$. The token-level confidence is measured as the maximum probability over the vocabulary:

$$c_t = \max\left(\text{softmax}(\mathbf{l}_t^{\text{LAM}})\right) \tag{6}$$

A fixed confidence threshold τ determines the decision. τ in our setting is determined based on language-specific data availability, as detailed in Appendix A.8.

- If $c_t \geq \tau \rightarrow$ Decode with LAM.
- If $c_t < \tau \rightarrow$ Contrastive decode with the aPTM.

This dynamic gating balances the fluency of the LAM with the factual reliability of the aPTM.

3.3 Contrastive Decoding with Top-K Masking and Layer-wise Contrast

When contrastive decoding is triggered, LGCD first computes logits of aPTM $\mathbf{l}_t^{\text{aPTM}}$ approximated with LoRA:

$$\mathbf{l}_t^{\text{aPTM}} = \mathbf{l}_t^{\text{LAM}} + \text{LoRA}(\Delta W_\ell, \mathbf{h}_t^{\text{LAM}}) \tag{7}$$

To prevent contrastive decoding from selecting tokens with low probabilities across both models, LGCD applies Top-K masking, considering only the K most probable tokens from $\mathbf{l}_t^{\text{LAM}}$:

$$\mathcal{T}_K = \text{TopK}(\mathbf{l}_t^{\text{LAM}}, K) \tag{8}$$

The contrastive logits are computed as:

$$\mathbf{l}_t^{\text{contrast}}[i] = \begin{cases} \text{if } i \in \mathcal{T}_K: & \mathbf{l}_t^{\text{LAM}}[i] + \beta \cdot \left(\mathbf{l}_t^{\text{aPTM}}[i] \quad - \alpha \cdot \mathbf{l}_t^{\text{LAM}}[i]\right) \\ \text{otherwise:} & -\infty \end{cases} \tag{9}$$

where β is a hyperparameter controlling the overall contrastive weighting and $\alpha \in [0, 1]$ controls the degree of down-weighting applied to the LAM logits within the correction term.

This formulation goes beyond standard contrastive decoding by introducing a correction term $\left(\mathbf{l}_t^{\text{aPTM}}[i] - \alpha \cdot \mathbf{l}_t^{\text{LAM}}[i]\right)$ that prioritizes the aPTM's factual knowledge while gently penalizing potentially overconfident LAM predictions. By tuning α, we modulate the LAM's influence without abruptly overriding it. Hyperparameter details are provided in Appendix A.4, and for completeness we include the full decoding algorithm and pseudocode in Appendix A.2.

4 EXPERIMENTAL SETUP

We evaluate our model on two task types—multiple-choice QA and long-form generation—across nine target languages: Chinese (zh), German (de), Portuguese (pt), Arabic (ar), Persian (fa), Japanese (ja), Korean (ko), Indonesian (id), and Swahili (sw). Unless otherwise noted, all experiments are conducted using 12 models.

4.1 MULTIPLE-CHOICE QA

Global MMLU. To assess multilingual factual understanding, we use Global MMLU (Singh et al., 2024), a culturally-aware extension of MMLU with 14K curated questions spanning 57 subjects in 42 languages. We report accuracy per language in zero- and five-shot settings for nine target langueages.

Multilingual TruthfulQA. We use the MC1 version of TruthfulQA (Dac Lai et al., 2023; Lin et al., 2021) across 31 languages. The number of questions per language varies, with most languages containing at least 700 items. We evaluate six models in zero-shot and five-shot settings.

4.2 LONG-FORM GENERATION

Medical QA. For high-stakes generative evaluation, we use multilingual medical QA datasets with expert-validated answers (Appendix A.5). Evaluation includes:

- **LLM-as-a-Judge:** GPT-4o performs pairwise comparisons with baselines (Li et al., 2023).
- **Human Evaluation:** For 12 LAMs, we sample 20 questions each (240 total). Outputs are translated to English and rated by three experts on fluency, coherence, specificity, and factuality. Final labels (*Win*/*Tie*/*Lose*) are based on majority vote.

Multi-FAct. We evaluate factuality with Multi-FAct (Shafayat et al., 2024), which uses FActScore (Min et al., 2023) to decompose generations into atomic facts and verify them against trusted sources in multilingual settings.

4.3 MODELS & BASELINES

Models. We evaluate 12 publicly available LAMs from Hugging Face, each specialized for one of the 9 target languages (*zh, de, pt, ar, fa, ja, ko, id, sw*). All models are based on multilingual LLM backbones (mainly LLaMA-3 variants) and further adapted via CPT, instruction tuning, or both. Model selection was guided by language specificity, public availability, and community engagement (e.g., download count, active maintenance). Table 1 summarizes model specifications.

Lang.	Model	CPT	Instr. tuning
zh	shenzhi-wang/Llama3-8B-Chinese-Chat	✓	✓
zh	hfl/llama-3-chinese-8b-instruct	✓	✓
de	DiscoResearch/Llama3-DiscoLeo-Instruct-8B-v0.1	✓	✓
pt	rhaymison/gemma-portuguese-luana-2b	✗	✓
ar	MohamedRashad/Arabic-Orpo-Llama-3-8B-Instruct	✗	✓
fa	PartAI/Dorna-Llama3-8B-Instruct	✗	✓
ja	tokyotech-llm/Llama-3-Swallow-8B-Instruct-v0.1	✓	✓
ja	elyza/Llama-3-ELYZA-JP-8B	✓	✓
ko	KISTI-KONI/KONI-Llama3-8B-Instruct-20240729	✓	✓
ko	MLP-KTLim/llama-3-Korean-Bllossom-8B	✓	✓
id	GoToCompany/llama3-8b-cpt-sahabatai-v1-instruct	✓	✓
sw	Jacaranda/UlizaLlama3	✓	✓

Table 1: LAMs used in our experiments. "✓" indicates application of CPT and/or instruction tuning.

Baselines. We compare LGCD against Nucleus Sampling (NS), DoLa (Chuang et al., 2024), TIES (Yadav et al., 2023), and SLERP (Shoemake, 1985)[1]. This set covers widely used decoding methods and model-merging approaches, enabling a comprehensive assessment of LGCD.

5 EXPERIMENTAL RESULTS

We evaluate LGCD on two multilingual multiple-choice QA benchmarks—Global MMLU and Multilingual TruthfulQA—under zero-shot and five-shot settings. Comparisons include decoding-time baselines (DoLa), model merging methods (TIES, SLERP), and standard nucleus sampling (NS) for both pretrained (PTM) and language-adapted models (LAM).

[1] https://github.com/Digitous/LLM-SLERP-Merge

Lang.	Model	0-shot						5-shot					
		PTM	LAM	DoLa	TIES	SLERP	LGCD	PTM	LAM	DoLa	TIES	SLERP	LGCD
zh	hfl/llama-3-chinese-8b-instruct	0.494	0.466	0.467	0.502	0.494	**0.519**	0.532	0.515	0.514	0.536	0.532	**0.543**
zh	shenzhi-wang/Llama3-8B-Chinese-Chat	0.494	0.500	0.500	0.498	0.494	**0.502**	0.532	**0.543**	0.542	0.538	0.532	0.543
de	DiscoResearch/Llama3-DiscoLeo-Instruct-8B-v0.1	0.514	0.486	0.492	0.540	0.514	**0.546**	0.558	0.548	0.550	0.566	**0.584**	0.574
pt	rhaymison/gemma-portuguese-luana-2b	0.353	0.316	0.278	**0.357**	0.353	**0.357**	0.325	0.316	0.298	**0.330**	0.325	0.324
ar	MohamedRashad/Arabic-Orpo-Llama-3-8B-Instruct	0.425	0.430	0.427	0.425	0.441	**0.465**	0.467	0.471	0.473	0.467	**0.483**	0.481
fa	PartAI/Dorna-Llama3-8B-Instruct	**0.424**	0.423	0.424	0.423	0.424	0.423	0.465	0.466	**0.468**	0.466	0.465	0.466
ja	tokyotech-llm/Llama-3-Swallow-8B-Instruct-v0.1	0.481	0.478	0.479	0.462	0.456	**0.507**	0.481	**0.527**	**0.527**	0.525	0.481	0.525
ja	elyza/Llama-3-ELYZA-JP-8B	0.481	0.473	0.473	0.466	0.456	**0.509**	0.481	0.503	0.503	0.513	0.481	**0.528**
ko	KISTI-KONI/KONI-Llama3-8B-Instruct-20240729	0.437	0.445	0.459	0.445	0.437	**0.490**	0.481	0.495	0.495	0.509	0.481	**0.511**
ko	MLP-KTLim/llama-3-Korean-Bllossom-8B	0.437	0.376	0.378	0.435	0.437	**0.479**	0.481	0.447	0.448	0.485	0.481	**0.501**
id	GoToCompany/llama3-8b-cpt-sahabatai-v1-instruct	0.476	0.530	**0.531**	0.486	0.476	0.527	0.529	0.576	**0.577**	0.530	0.568	0.566
sw	Jacaranda/UlizaLlama3	0.359	0.362	0.363	0.355	0.359	**0.399**	0.388	0.367	0.372	0.390	**0.432**	0.409
	Average	0.448	0.441	0.439	0.449	0.445	**0.477**	0.477	0.481	0.481	0.488	0.487	**0.498**

Table 2: Evaluation accuracy of 12 models on Global MMLU benchmark under 0-shot and 5-shot settings using various decoding and merging strategies. PTM refers to the performance of the pretrained Model, while LAM denotes the language-adapted model, both evaluated using Nucleus Sampling (NS). DoLa represents the results when applying the DoLa decoding strategy to the LAM.

Lang.	Model	0-shot						5-shot					
		PTM	LAM	DoLa	TIES	SLERP	LGCD	PTM	LAM	DoLa	TIES	SLERP	LGCD
zh	hfl/llama-3-chinese-8b-instruct	0.349	**0.353**	0.312	0.352	0.335	0.352	0.379	0.390	0.392	0.381	0.363	**0.471**
zh	shenzhi-wang/Llama3-8B-Chinese-Chat	0.349	**0.363**	0.326	0.357	**0.363**	0.357	0.379	0.391	0.360	0.387	0.400	**0.484**
de	DiscoResearch/Llama3-DiscoLeo-Instruct-8B-v0.1	0.317	0.343	0.320	0.325	0.330	**0.382**	0.367	0.367	0.354	0.373	**0.393**	0.391
pt	rhaymison/gemma-portuguese-luana-2b	0.272	0.301	0.268	0.283	0.302	**0.409**	0.319	0.325	0.299	0.322	0.321	**0.431**
ar	MohamedRashad/Arabic-Orpo-Llama-3-8B-Instruct	0.325	0.308	0.331	0.326	0.318	**0.400**	0.376	0.360	0.323	0.383	0.365	**0.426**
id	GoToCompany/llama3-8b-cpt-sahabatai-v1-instruct	0.329	0.334	0.334	0.341	0.334	**0.353**	0.378	0.370	0.369	0.373	0.365	**0.406**
	Average	0.323	0.334	0.315	0.331	0.330	**0.376**	0.366	0.367	0.349	0.370	0.368	**0.435**

Table 3: Evaluation accuracy of 6 models on multilingual TruthfulQA benchmark under 0-shot and 5-shot settings using different merging and decoding strategies.

5.1 GLOBAL MMLU

Table 2 shows accuracy across 12 LAMs covering 9 languages. LGCD achieves the best average performance in both zero-shot and five-shot settings, outperforming all decoding and merging baselines.

In the zero-shot setting, relative to the LAM, LGCD improves accuracy in 10 of 12 cases. Gains are largest where the LAM lags the original PTM (Korean: +4.5–10.3 pp; German: +6.0 pp; Japanese: +2.9–3.6 pp; Portuguese: +4.1 pp), indicating that LGCD effectively recovers knowledge lost during adaptation.

In contrast, in the five-shot setting, LGCD continues to provide consistent improvements, especially in high-resource languages such as Chinese and German. This indicates that LGCD not only recovers forgotten knowledge but also scales robustly with richer context, effectively balancing domain adaptation and pretrained factuality across diverse conditions.

Overall, LGCD consistently improves multilingual QA performance, showing robustness across culturally diverse and knowledge-heavy questions where language adaptation typically disrupts general knowledge.

5.2 MULTILINGUAL TRUTHFULQA

Results on Multilingual TruthfulQA (Table 3) further validate LGCD's ability to enhance factuality. In the zero-shot setting, LGCD achieves the highest average accuracy, outperforming both decoding and merging baselines across all evaluated languages.

LGCD demonstrates consistent improvements across all tested languages, with particularly notable gains in Portuguese (+10.8 pp), Arabic (+9.2 pp), and German (+3.9 pp). These improvements hold in the five-shot setting as well, where LGCD shows even larger gains, reinforcing the method's robustness across different prompting regimes.

These results confirm that LGCD effectively resists plausible but incorrect generations, excelling in settings in TruthfulQA that probe a model's ability to distinguish truth from commonly held misconceptions.

Figure 2: Pairwise comparison results of our model vs. baselines (evaluated by GPT-4o)

5.3 Long-form Medical QA

We further evaluate LGCD in high-stakes, long-form medical question answering tasks across 12 LAMs. Each model is tested in its target language using domain-specific medical QA datasets. Evaluation is conducted via two complementary methods: (1) GPT-4o as an automatic judge performing pairwise comparisons, and (2) expert human preference evaluations on fluency, coherence, and factual correctness.

LLM-as-a-Judge Evaluation. Figure 2 shows GPT-4o-based preference comparisons. LGCD achieves the highest win rates in most languages, outperforming PTM, LAM, and all decoding or merging baselines. On average, LGCD is preferred over PTM in 63.1% of cases, over LAM in 53.5%, and over DoLa (53.8%) and SLERP (51.9%), while performing competitively with TIES (65.3%).

Human Preference Evaluation. We further assess LGCD through human preference evaluation. As shown in Figure 3, LGCD is consistently favored across baselines, achieving higher win rates than PTM (52.0%), DoLA (45.8%), and SLERP (45.8%). These results reinforce LGCD's ability to generate fluent and factually grounded answers in complex medical domains. Despite cross-lingual pooling, the consistent preference trend across models suggests that LGCD offers a robust and training-free alternative to improve factuality in long-form, domain-specific generation.

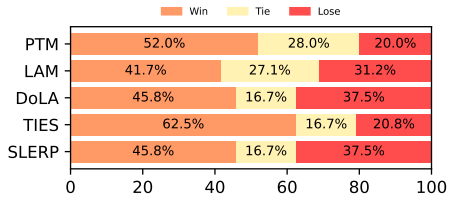

Figure 3: Human preference comparison between our model and baseline models

Contrastive Usage Analysis Figure 4 shows the proportion of tokens decoded by the LAM (blue) and via contrastive decoding (red) as the confidence threshold τ varies. The yellow band indicates the threshold selected for each model.

In higher-resource languages (e.g., zh, de, pt, ar, fa), models adopt higher thresholds (typically $\tau = 0.7$–0.8), leading to a sharp increase in contrastive usage. This reflects the stronger performance of the pretrained model in these languages, allowing LGCD to revise uncertain predictions more effectively.

In contrast, for lower-resource languages (e.g., ja, ko, id, sw), LAM usage remains high even at high thresholds. These models tend to produce overconfident outputs, possibly due to limited token coverage during adaptation. Since the PTM is also less reliable in these settings, LGCD uses lower thresholds to favor the LAM, which yields better results despite overconfidence.

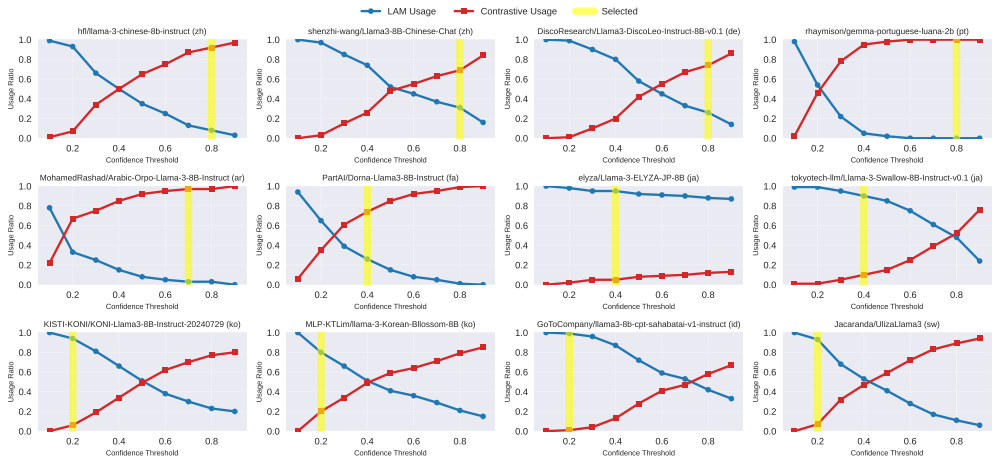

Figure 4: Contrastive decoding behavior in LGCD across 12 LAMs. For each model, we plot the token-level usage ratio of the base LAM (blue) and contrastive candidate (red) as a function of confidence threshold τ. The yellow vertical line marks the threshold selected for that model. LGCD dynamically adjusts model usage based on token uncertainty, striking a balance between domain alignment and factual recall.

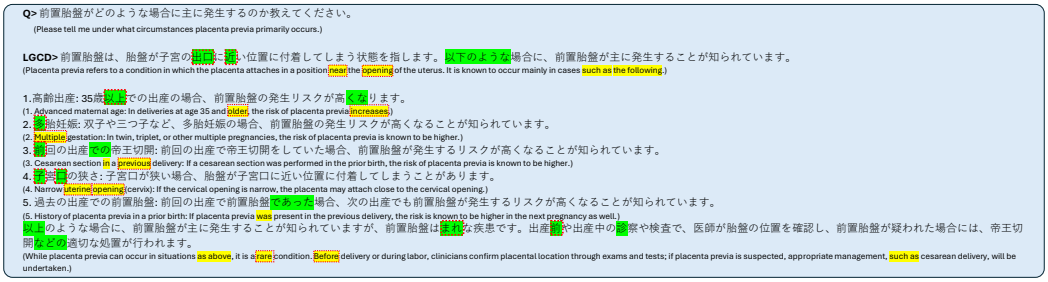

Figure 5: Token-level view on a Japanese medical QA example (elyza/Llama-3-ELYZA-JP-8B). LGCD output in Japanese (top) with English alignment (bottom). Tokens highlighted in green mark time-steps where LGCD switched on contrastive decoding; their aligned tokens in the translation are highlighted in yellow. Tokens outlined in red indicate decisive positions that steered the continuation toward the factual answer.

To probe where LGCD's gains come from in long-form generation, we analyze named-entity behavior under a general Named Entity Recognition (NER) schema and compare LGCD to the underlying LAM. Further details are provided in the Appendix A.7. Figure 6 summarizes two signals: (i) average number of entities extracted per output and (ii) Jaccard overlap between the entity sets from LGCD and the LAM. Across three representative LAMs (Chinese, Japanese, Indonesian), LGCD consistently produces more entities than the LAM while the set overlap remains low–moderate ($\approx$ 1–16%). This pattern suggests LGCD is not merely echoing the LAM's choices but is adding complementary, likely factual, mentions that the LAM omits. Linking back to the Contrastive Usage Analysis, these results explain why factuality can improve even when overall contrastive usage is low—e.g., in Japanese (elyza/Llama-3-ELYZA-JP-8B) and Indonesian (GoToCompany/llama3-8b-cpt-sahabatai-v1-instruct).

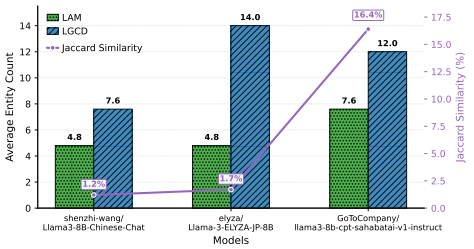

Figure 6: Average entity count (left) and Jaccard similarity (right) between LGCD and LAM outputs for Chinese, Japanese, and Indonesian models

Lang.	Model	LAM	LGCD	Δ
zh	hfl/llama-3-chinese-8b-instruct	0.260	0.246	−0.014
zh	shenzhi-wang/Llama3-8B-Chinese-Chat	0.229	**0.313**	+0.084
de	DiscoResearch/Llama3-DiscoLeo-Instruct-8B-v0.1	0.387	**0.520**	+0.133
pt	rhaymison/gemma-portuguese-luana-2b	**0.288**	0.267	−0.021
ja	tokyotech-llm/Llama-3-Swallow-8B-Instruct-v0.1	**0.267**	0.197	−0.070
ja	elyza/Llama-3-ELYZA-JP-8B	0.298	**0.302**	+0.004
ko	KISTI-KONI/KONI-Llama3-8B-Instruct-20240729	0.196	**0.218**	+0.022
ko	MLP-KTLim/llama-3-Korean-Bllossom-8B	0.189	**0.196**	+0.007
id	GoToCompany/llama3-8b-cpt-sahabatai-v1-instruct	0.334	**0.547**	+0.212
	Average	0.272	**0.312**	**+0.040**

Table 4: Comparison of Multi-FAct score between LAM and LGCD. Models with a LAM score below 0.05 were excluded, resulting in 9 evaluated models.

Figure 5 further illustrates that LGCD intervenes only sparsely, yet the activated gates coincide with tokens that are not only entity mentions but also those carrying decisive factual content. Despite their small number, these targeted interventions are sufficient to steer the generation toward factual answers. A direct comparison of LAM and LGCD outputs for the Japanese medical QA example is provided in the Appendix A.9.

5.4 Multi-FAct

On the Multi-FAct benchmark, LGCD improves factual consistency in 6 of 9 models, with a +0.04 average gain over LAMs (Table 4). Notable gains in Indonesian, German and Korean confirm LGCD's effectiveness in enhancing factuality for long-form generation.

5.5 Throughput

Table 5 presents the decoding speed of different generation strategies measured on a single A100 GPU. Results are averaged over responses to 100 questions using the `hfl/llama-3-chinese-8b-instruct` model. Among baselines, greedy search achieves the highest throughput (19.21 tokens/sec), followed by nucleus sampling (17.47). Contrastive Search is slower (11.87) due to the similarity penalty over top-k candidates, though it runs in a single forward pass without reranking. LGCD slows decoding by querying the aPTM when the LAM lacks confidence. However, the overhead varies with the confidence threshold τ: lower thresholds (e.g., LGCD-0.2) lead to fewer contrastive decisions and thus higher throughput (14.37), while higher τ (e.g., LGCD-0.8) leads to more aggressive factual intervention and slower speed (10.22). Notably, LGCD-0.2 approaches the speed of DoLa (16.81), demonstrating that factual calibration can be achieved without substantial efficiency loss when appropriately tuned. This tunability enables LGCD to balance quality and speed for practical deployment.

Decoding Strategy	Throughput (Token/s)
Greedy search	19.21
Nucleus sampling	17.47
Contrastive search	11.87
DoLa	16.81
LGCD-0.2	14.37
LGCD-0.4	10.32
LGCD-0.6	10.32
LGCD-0.8	10.22

Table 5: Average decoding throughput for each strategy. For LGCD-τ, the numeric suffix denotes the confidence threshold τ used in decoding.

6 Conclusion

We present LGCD, a novel training-free method that enhances the factuality of language-adapted LLMs by dynamically injecting pretrained knowledge during inference. By extracting factual signals from FFN layers via LoRA-based decomposition and applying contrastive decoding conditioned on token-level confidence, LGCD effectively mitigates catastrophic forgetting without retraining or access to pretraining data. Extensive multilingual experiments across multiple-choice QA and long-form generation tasks demonstrate that LGCD consistently improves factual accuracy over both decoding and model-merging baselines, offering a practical, scalable solution for factuality preservation in domain-specialized LLMs.

References

Julien Abadji, Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. Ungoliant: An optimized pipeline for the generation of a very large-scale multilingual web corpus. Proceedings

of the Workshop on Challenges in the Management of Large Corpora (CMLC-9) 2021. Limerick, 12 July 2021 (Online-Event), pp. 1 – 9, Mannheim, 2021. Leibniz-Institut für Deutsche Sprache. doi: 10.14618/ids-pub-10468. URL https://nbn-resolving.org/urn:nbn:de:bsz:mh39-104688.

Julien Abadji, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot. Towards a Cleaner Document-Oriented Multilingual Crawled Corpus. *arXiv e-prints*, art. arXiv:2201.06642, January 2022.

Isaac Caswell, Julia Kreutzer, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Javier Ortiz Suárez, Iroro Orife, Kelechi Ogueji, Rubungo Andre Niyongabo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. Quality at a Glance: An Audit of Web-Crawled Multilingual Datasets. *arXiv e-prints*, art. arXiv:2103.12028, March 2021.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R. Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Th6NyL07na.

Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A Rossi, and Thien Huu Nguyen. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. *arXiv e-prints*, pp. arXiv–2307, 2023.

Damai Dai, Wenbin Jiang, Qingxiu Dong, Yajuan Lyu, and Zhifang Sui. Neural knowledge bank for pretrained transformers. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pp. 772–783. Springer, 2023.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. *arXiv preprint arXiv:2012.14913*, 2020.

Jiawei Gu, Zacc Yang, Chuanghao Ding, Rui Zhao, and Fei Tan. Cmr scaling law: Predicting critical mixture ratios for continual pre-training of language models. *arXiv preprint arXiv:2407.17467*, 2024.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*, 2020.

Jinghan He, Haiyun Guo, Kuan Zhu, Zihan Zhao, Ming Tang, and Jinqiao Wang. Seekr: Selective attention-guided knowledge retention for continual learning of large language models. *arXiv preprint arXiv:2411.06171*, 2024.

Jianheng Huang, Leyang Cui, Ante Wang, Chengyi Yang, Xinting Liao, Linfeng Song, Junfeng Yao, and Jinsong Su. Mitigating catastrophic forgetting in large language models with self-synthesized rehearsal. *arXiv preprint arXiv:2403.01244*, 2024.

Quzhe Huang, Mingxu Tao, Chen Zhang, Zhenwei An, Cong Jiang, Zhibin Chen, Zirui Wu, and Yansong Feng. Lawyer llama technical report. *arXiv preprint arXiv:2305.15062*, 2023.

Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*, 2020.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38, 2023.

Damjan Kalajdzievski. Scaling laws for forgetting when fine-tuning large language models. *arXiv preprint arXiv:2401.05605*, 2024.

Chen-An Li and Hung-Yi Lee. Examining forgetting in continual pre-training of aligned large language models. *arXiv preprint arXiv:2401.03129*, 2024.

Hongyu Li, Liang Ding, Meng Fang, and Dacheng Tao. Revisiting catastrophic forgetting in large language model tuning. *arXiv preprint arXiv:2406.04836*, 2024.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. From quantity to quality: Boosting llm performance with self-guided data selection for instruction tuning. *arXiv preprint arXiv:2308.12032*, 2023.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods, 2021.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*, 2023.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*, 2023.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*, 2024.

Zihan Qiu, Zeyu Huang, Youcheng Huang, and Jie Fu. Empirical study on updating key-value memories in transformer feed-forward layers. *arXiv preprint arXiv:2402.12233*, 2024.

Sheikh Shafayat, Eunsu Kim, Juhyun Oh, and Alice Oh. Multi-fact: Assessing multilingual llms' multi-regional knowledge using factscore, 2024.

Ken Shoemake. Animating rotation with quaternion curves. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pp. 245–254, 1985.

Shivalika Singh, Angelika Romanou, Clémentine Fourrier, David I Adelani, Jian Gang Ngui, Daniel Vila-Suero, Peerat Limkonchotiwat, Kelly Marchisio, Wei Qi Leong, Yosephine Susanto, et al. Global mmlu: Understanding and addressing cultural and linguistic biases in multilingual evaluation. *arXiv preprint arXiv:2412.03304*, 2024.

Yixuan Su, Tian Lan, Yan Wang, Dani Yogatama, Lingpeng Kong, and Nigel Collier. A contrastive framework for neural text generation. *Advances in Neural Information Processing Systems*, 35: 21548–21561, 2022.

Anh-Dung Vo, Minseong Jung, Wonbeen Lee, and Daewoo Choi. Redwhale: An adapted korean llm through efficient continual pretraining. *arXiv preprint arXiv:2408.11294*, 2024.

Cunxiang Wang, Haofei Yu, and Yue Zhang. Rfid: Towards rational fusion-in-decoder for open-domain question answering. *arXiv preprint arXiv:2305.17041*, 2023a.

Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. Orthogonal subspace learning for language model continual learning. *arXiv preprint arXiv:2310.14152*, 2023b.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Resolving interference when merging models. *arXiv preprint arXiv:2306.01708*, 1, 2023.

Di Zhang, Wei Liu, Qian Tan, Jingdan Chen, Hang Yan, Yuliang Yan, Jiatong Li, Weiran Huang, Xiangyu Yue, Dongzhan Zhou, et al. Chemllm: A chemical large language model. *arXiv preprint arXiv:2402.06852*, 2024.

A APPENDIX

A.1 LLM USAGE DISCLOSURE

We used LLMs in the following limited ways to aid our research process:

- **Writing Assistance**: ChatGPT-5 was used to polish and refine writing, including grammar correction, translation for clarity, and LaTeX table formatting support. All generated or suggested content was carefully reviewed and edited by the authors before inclusion.
- **Evaluation (LLM-as-Judge)**: GPT-4o was employed as an automatic evaluator for long-form medical QA performance, and internally within the Multi-FAct benchmark, following established "LLM-as-judge" protocols.
- **Code Debugging**: ChatGPT-5 was used in a supporting role to debug code. The authors independently verified the correctness of all outputs.

The LLMs did not contribute to research ideation or the design of experiments. All final content and claims in the paper remain the responsibility of the authors.

A.2 DECODING PROCESS

Algorithm 1 summarizes the complete LGCD procedure. At each timestep t, the framework evaluates token-level confidence c_t from the LAM. When $c_t \geq \tau$, decoding proceeds with the LAM alone using Top-K sampling. When $c_t < \tau$, indicating potential factual uncertainty, LGCD activates the contrastive mechanism: (1) computing LoRA-approximated pretrained logits, (2) dynamically gating based on token-level confidence, and (3) applying contrastive weighting within the Top-K space. This confidence-driven approach enables LGCD to dynamically balance domain fluency and factual accuracy without requiring model retraining, making it particularly suitable for scenarios where language adaptation may compromise general knowledge.

Algorithm 1 LoRA-Gated Contrastive Decoding (LGCD)

Require: Language-Adapted Model M_{LAM}, Approximated Pretrained Model M_{aPTM} (for ΔW_ℓ), confidence threshold τ, Top-K size K, contrastive weighting hyperparameter β
Ensure: Generated output sequence $\mathcal{Y}$

Initialize $\mathcal{Y} = [\,]$
while $\mathcal{Y}[t] \neq$ <eos>, $t = 1, 2, \ldots$ **do**
  Compute LAM logits $\mathbf{l}_t^{\text{LAM}}$
  Compute token-level confidence c_t:
    $c_t = \max(\text{softmax}(\mathbf{l}_t^{\text{LAM}}))$
  if $c_t \geq \tau$ **then**
    Apply Top-K masking: $\mathcal{T}_K = \texttt{TopK}(\mathbf{l}_t^{\text{LAM}}, K)$
    Select next token y_t by sampling from the masked $\mathbf{l}_t^{\text{LAM}}$
  else
    Compute LoRA-approximated pretrained logits:
      $\mathbf{l}_t^{\text{aPTM}} = \mathbf{l}_t^{\text{LAM}} + \texttt{LoRA}(\Delta W_\ell, \mathbf{h}_t^{\text{LAM}})$
    Apply Top-K masking: $\mathcal{T}_K = \texttt{TopK}(\mathbf{l}_t^{\text{LAM}}, K)$
    Initialize $\mathbf{l}_t^{\text{contrast}} = [-\infty, \ldots, -\infty] \in \mathbb{R}^{\text{vocab\_size}}$
    for each token i in vocabulary **do**
      if $i \in \mathcal{T}_K$ **then**
        Compute contrastive logit:
          $\mathbf{l}_t^{\text{contrast}}[i] = \mathbf{l}_t^{\text{LAM}}[i] + \beta \cdot \big(\mathbf{l}_t^{\text{aPTM}}[i]$
            $-0.1 \cdot \mathbf{l}_t^{\text{LAM}}[i]\big)$
      else
        $\mathbf{l}_t^{\text{contrast}}[i] = -\infty$
      end if
    end for
    Select next token y_t by sampling from $\mathbf{l}_t^{\text{contrast}}$
  end if
  Append y_t to output sequence $\mathcal{Y}$
end while
return $\mathcal{Y}$

A.3 LoRA-Based PTM Approximation: FFN vs QV vs All-Layer Comparison

LGCD relies on approximating the PTM by applying LoRA-based updates to the LAM, using low-rank matrices derived from the difference between PTM and LAM parameters. In our primary design, this approximation targets only the FFN layers, based on prior research showing that FFNs encode core factual knowledge in LLMs (Geva et al., 2020; Qiu et al., 2024; Dai et al., 2023).

To assess the importance of this design choice, we compare three strategies for selecting which layers to approximate via LoRA during the offline distillation step:

- **QV-only:** Apply LoRA decomposition only to the attention projection matrices (Q, V).
- **FFN-only:** Apply LoRA only to FFN layers (our default).
- **All Layers:** Apply LoRA to both FFN and attention projection layers.

Table 6 shows the performance of LGCD using each variant on the Global MMLU benchmark under the zero-shot setting. Results show that FFN-only approximation consistently achieves the best or comparable performance, while QV-only performs worse on nearly all models. Approximating all layers introduces marginal gains in a few cases but often results in unstable or degraded performance, suggesting that unnecessary modification of attention layers may introduce noise. These findings empirically validate our FFN-only design as both effective and efficient for factuality-oriented approximation of PTMs.

Lang.	Model	QV-only	FFN-only	All Layers
zh	hfl/llama-3-chinese-8b-instruct	0.485	0.519	**0.520**
zh	shenzhi-wang/Llama3-8B-Chinese-Chat	0.500	0.502	**0.503**
de	DiscoResearch/Llama3-DiscoLeo-Instruct-8B-v0.1	0.520	**0.546**	0.544
pt	rhaymison/gemma-portuguese-luana-2b	0.330	**0.357**	0.340
ar	MohamedRashad/Arabic-Orpo-Llama-3-8B-Instruct	0.460	**0.465**	0.460
fa	PartAI/Dorna-Llama3-8B-Instruct	**0.423**	**0.423**	**0.423**
ja	tokyotech-llm/Llama-3-Swallow-8B-Instruct-v0.1	0.499	**0.507**	0.503
ja	elyza/Llama-3-ELYZA-JP-8B	0.499	**0.509**	**0.509**
ko	KISTI-KONI/KONI-Llama3-8B-Instruct-20240729	0.484	**0.490**	0.484
ko	MLP-KTLim/llama-3-Korean-Bllossom-8B	0.471	**0.479**	0.461
id	GoToCompany/llama3-8b-cpt-sahabatai-v1-instruct	**0.533**	0.527	0.528
sw	Jacaranda/UlizaLlama3	0.375	**0.399**	0.395
Average		0.465	**0.477**	0.472

Table 6: Ablation study on LoRA-based PTM approximation: comparing factual accuracy on Global MMLU (0-shot) when contrastive decoding uses knowledge reconstructed from different layer subsets.

A.4 Configurations for LGCD

LGCD is configured with task-specific decoding strategies to address the differing demands of multiple-choice QA and long-form QA. While both settings use the same LoRA-based decomposition applied to all FFN layers with a fixed rank of 32, the confidence threshold and usage of contrastive decoding vary by task.

For **multiple-choice QA**, we use a fixed confidence threshold of $\tau = 0.9$ across all languages. This high threshold leads to frequent activation of contrastive decoding, ensuring that the model does not rely solely on the LAM but actively considers both the LAM and the PTM when scoring candidate answers. In this setup, LGCD functions not merely as a fallback for low-confidence predictions, but as a mechanism to integrate knowledge from both models during answer selection, enhancing factual reliability without requiring additional training. Decoding uses top-k sampling ($k = 100$) with temperature 0.7 and top-$p = 0.9$, and all contrastive updates are scaled with $\beta = 1.0$.

For **long-form QA**, the decoding configuration remains identical in terms of sampling and contrastive scaling, but the confidence threshold τ is language-specific, calibrated according to resource availability (see Appendix A.8). This allows LGCD to adaptively determine when to apply con-

trastive decoding based on the reliability of the LAM for each language—more frequently for low-resource languages and conservatively for high-resource ones.

Component	Multiple-Choice QA	Long-Form QA
Temperature	0.7	0.7
Top-k	100	100
Top-p	0.9	0.9
Contrastive β	1.0	1.0
Contrastive α	0.1	0.1
LoRA (Rank / Scope)	32 / All FFN layers	32 / All FFN layers
Confidence τ	Fixed (0.9)	Language-specific
Contrastive Use	Joint scoring	Factual correction

Table 7: LGCD decoding configurations for multiple-choice vs. long-form QA.

A.5 Evaluation Datasets for Multilingual Medical Question Answering

This appendix provides a detailed account of the datasets employed to evaluate the multilingual medical question answering capabilities of the models discussed in this study. We have curated a collection of open-ended medical QA datasets across various languages, drawing from both established benchmarks and language-specific resources.

Our evaluation utilizes open-ended medical question answering data. For several languages, the Healthcare QA task from the AIR-Bench 24.05 benchmark (`https://github.com/AIR-Bench/AIR-Bench`) serves as the primary data source. This task comprises questions within the medical domain designed to assess a model's ability to provide informative responses. Specifically, the original Healthcare QA data from AIR-Bench was directly used for evaluation in Arabic and German.

For a broader linguistic evaluation where native high-quality long-form medical QA datasets were not readily available, we constructed evaluation sets by translating the English version of the Healthcare QA task from AIR-Bench. These translations were systematically performed using Google Translate. This methodology was applied to generate the evaluation datasets for Persian, Indonesian, Japanese, Portuguese, and Swahili. This approach enables a comparative analysis based on a consistent source structure across these languages, although with potential translation artifacts.

In addition to the AIR-Bench based datasets, we incorporated independent, language-specific medical QA datasets for Korean and Chinese:

- **Korean:** We used the `ChuGyouk/GenMedGPT-5k-ko` dataset (`https://huggingface.co/datasets/ChuGyouk/GenMedGPT-5k-ko`) for long-form medical QA in Korean. This dataset, containing approximately 5,000 question-answer pairs, is a Korean translation, performed using DeepL, of medical QA data sourced from `https://github.com/Kent0n-Li/ChatDoctor`.
- **Chinese:** For Chinese medical QA, we utilize the `cMedQA2` dataset (`https://github.com/zhangsheng93/cMedQA2`). This dataset is a dedicated resource for medical QA in Chinese, comprising a collection of questions and corresponding expert answers within the medical domain.

A.6 Baseline Configuration Details

We compare LGCD against both decoding-based and model-merging baselines. All decoding strategies use the same chat template with `max_new_tokens = 2048`.

Decoding-Based Baselines

We evaluate five decoding strategies: greedy decoding (GS), contrastive search (CS), nucleus sampling (NS), and DoLa. All hyperparameters follow either the original paper or Hugging Face implementation standards.

In particular, DoLa adopts a task-sensitive contrastive scheme: it uses higher transformer layers for multiple-choice QA and lower layers for long-form generation, following prior findings on factual calibration via depth selection.

Method	Hyperparameters
GS	Hugging Face default (greedy decoding)
CS	`penalty_alpha = 0.6, top_k = 4`
NS	`do_sample = True, temperature = 0.7, top_p = 0.9`
DoLa	`do_sample = False`
	`dola_layers = "high"` (Multiple-Choice QA)
	`dola_layers = "low"` (Long-Form QA)

Table 8: Decoding hyperparameter settings for baseline methods.

MODEL-MERGING BASELINES

We further compare LGCD with two weight-space integration baselines—**SLERP** and **TIES**—implemented using the `mergekit` framework. Both methods perform symmetric merging between the PTM and the LAM with the following common settings:

- Merge ratio: 0.5 (LAM) : 0.5 (PTM)
- Merge range: all transformer layers
- Data type: bfloat16
- Base model: LAM
- Other parameters: mergekit defaults

A.7 NER-BASED FACTUALITY PROBE

NER extraction. We measure entity-level differences between LGCD and the baseline LAM by applying a general-purpose NER extractor to every generated output. Specifically, we use GPT-4o as the extractor. The schema is based on coarse-grained categories commonly used in NER (e.g., `PERSON`, `ORGANIZATION`, `LOCATION`, `DATE`, `NUMBER`), and we extended it with `DISEASE` to capture domain-relevant mentions. In total, we use 13 categories. Each entity is returned with its surface form, category, and character offset.

Comparison metrics. For each output pair, we form entity sets E_{LGCD} and E_{LAM}. We compute Jaccard similarity $\frac{|E_{\text{LGCD}} \cap E_{\text{LAM}}|}{|E_{\text{LGCD}} \cup E_{\text{LAM}}|}$. We also track absolute entity counts per output. This setup allows us to test whether factuality gains from LGCD are reflected in increased coverage or correction of entity mentions beyond those produced by the base LAM.

A.8 LANGUAGE-SPECIFIC CONFIDENCE THRESHOLDS BASED ON DATA AVAILABILITY

Our decoding framework dynamically balances contributions between a language-adapted model and a pretrained multilingual backbone based on a token-level confidence threshold. When the model's token-level confidence falls below the threshold, we incorporate contrastive logits from the pretrained model to supplement or correct uncertain predictions.

Given that the quality and extent of training for language-adapted models are often opaque, especially in multilingual settings, we estimate data availability using the OSCAR 22.01 corpus Caswell et al. (2021); Abadji et al. (2022); Abadji et al. (2021) as a public proxy for the quantity of available web-scale text per language. Languages are categorized into three tiers—High, Medium, and Low—based on word count.

We assign lower thresholds (e.g., 0.1) to low-resource languages such as Korean (ko), Indonesian (id), and Swahili (sw), allowing the pretrained model's generalized knowledge to exert greater influence. Conversely, for high-resource languages like German (de) and Chinese (zh), which have abundant data, we set a higher threshold (e.g., 0.8), favoring the language-specific model. Medium-resource languages such as Japanese (ja) and Persian (fa) are assigned intermediate values (e.g.,

0.4). This approach ensures informed integration of pretrained knowledge proportional to resource availability.

Category	Languages (ISO code)	Word Count
High ($\geq$ 2B)	German (de)	46.8B
	Chinese (zh)	23.1B
	Portuguese (pt)	18.4B
Medium (0.5–2B)	Persian (fa)	6.4B
	Arabic (ar)	6.1B
	Japanese (ja)	5.6B
Low ($<$ 0.5B)	Korean (ko)	3.9B
	Indonesian (id)	2.0B
	Swahili (sw)	$\approx$7MB

Table 9: Language data categorization based on the public OSCAR 22.01 corpus.

A.9 Direct Comparison of LAM and LGCD Outputs

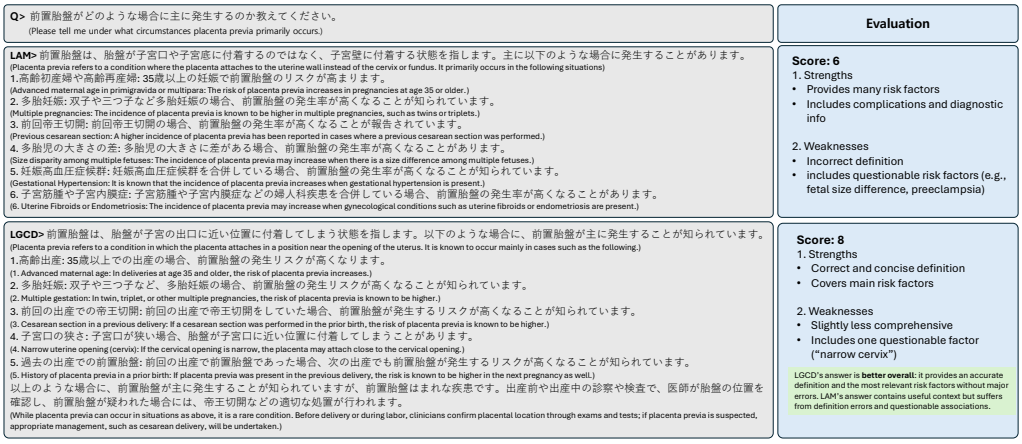

Figure 7: Direct comparison of LAM and LGCD outputs for a Japanese medical QA example (elyza/Llama-3-ELYZA-JP-8B) with evaluation