Multilingual Contrastive Decoding via Language-Agnostic Layers Skipping

Wenhao Zhu[1*], **Sizhe Liu**[1*], **Shujian Huang**[1], **Shuaijie She**[1], **Chris Wendler**[2], **Jiajun Chen**[1]
[1] National Key Laboratory for Novel Software Technology, Nanjing University [2] EPFL
zhuwh@smail.nju.edu.cn, liusz@smail.nju.edu.cn, shesj@smail.nju.edu.cn,
huangsj@nju.edu.cn, chris.wendler@epfl.ch, chenjj@nju.edu.cn

Abstract

Decoding by contrasting layers (DoLa), is designed to improve the generation quality of large language models (LLMs) by contrasting the prediction probabilities between an early exit output (amateur logits) and the final output (expert logits). However, we find that this approach does not work well on non-English tasks. Inspired by previous interpretability work on language transition during the model's forward pass, we discover that this issue arises from a language mismatch between early exit output and final output. In this work, we propose an improved contrastive decoding algorithm that is effective for diverse languages beyond English. To obtain more helpful amateur logits, we devise two strategies to skip a set of bottom, language-agnostic layers based on our preliminary analysis. Experimental results on multilingual reasoning benchmarks demonstrate that our proposed method outperforms previous contrastive decoding baselines and substantially improves LLM's chain-of-thought reasoning accuracy across 11 languages[1].

1 Introduction

Contrastive decoding (Li et al., 2023) presents a novel approach to enhance the text generation quality of large language models. At each inference step, contrastive decoding uses logits generated by an amateur model (usually small) to contrast with the output logits of an expert model (usually large). This reduces the probability of the expert model to make similar mistakes as the amateur model, thus making the generation content more logical and coherent (Li et al., 2023; O'Brien and Lewis, 2023; Zhao et al., 2024). To further eliminate the need of finding an extra amateur LLM, Chuang et al. (2023) propose DoLa, which uses the expert model's early exit output as amateur logits.

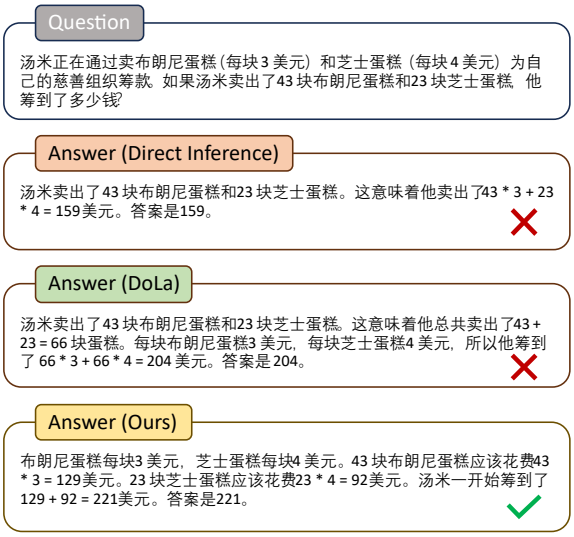

Figure 1: Illustration of the superiority of our proposed layer skipping contrastive decoding algorithm over direct inference and DoLa.

However, in this paper, we find that DoLa does not work well on non-English tasks. Inspired by the recent interpretability study by Wendler et al. (2024), which analyzes the language transitions during the forward pass, we identify that the issue with DoLa arises from the language mismatch between amateur logits and expert logits. Specifically, the early exit logits accumulate on English tokens even during non-English generation, thus failing to provide a helpful contrastive distribution for the expert model.

Contributions To obtain more helpful amateur logits, we propose an improved contrastive decoding algorithm by skipping a set of lower language-agnostic layers while preserving the computations in the upper transformer blocks (Figures 3). Specifically, we design two strategies to determine the positions for layer skipping: one based on heuristic rules, and the other based on entropy change.

Our experimental results on multilingual reason-

[*]Equal contribution.
[1]The project will be available at: https://github.com/NJUNLP/SkipLayerCD.

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 8775–8782
November 12-16, 2024

ing benchmarks MGSM show that our devised approach significantly outperforms the previous contrastive decoding approach DoLa, and improves the chain-of-thought reasoning accuracy of a group of open-source LLMs: LLaMA2 (Touvron et al., 2023a), LLaMA3 (Meta, 2024), Mistral (Jiang et al., 2023), etc., across 11 languages. The performance gap between our approach and DoLa on the multilingual benchmark also validates the findings about the language transition of intermediate decodings across the layers of LLMs by Wendler et al. (2024) and provides further insight into the working patterns of LLMs.

2 Background and Preliminary Analysis

In this section, we will briefly introduce the background of contrastive decoding and discuss why DoLa can not work well on non-English tasks.

2.1 Contrastive Decoding

While LLMs have shown impressive potential as foundation models (Touvron et al., 2023a; Jiang et al., 2023), they still easily make logical mistakes or generate hallucinations, especially in scenarios such as complex reasoning. To improve LLM's generation quality, Li et al. (2023) propose an effective decoding algorithm called contrastive decoding. This method uses the output logits from a small amateur model as a negative bias and subtracts this bias from the output logits of a large expert model at each inference step. But in practice, it is often hard to find a suitable amateur LLM that is smaller in size and shares the same vocabulary as the expert LLM. Therefore, Chuang et al. (2023) propose an amateur-free contrastive decoding method DoLa, which uses the early exit probabilities from the bottom layers as the amateur logits.

2.2 The Problem with Early Exit During Multilingual Generation

Despite DoLa's effectiveness on improving English generation, we discover that this approach does not perform well on non-English tasks (see Table 2). We find that this issue stems from a language mismatch between the early exit output and the final output. Figure 2 provides an empirical evidence for this observation[2] where we use the logit lens (nostalgebraist, 2020) to analyze each layer's output of Mistral-7B. We observe that Mistral does not generate tokens in the target language (Chinese, in this case) in the early exit output until it reaches the last few layers. This language mismatch fails to contribute to meaningful contrasting, thus leading to DoLa's shortcomings on non-English tasks.

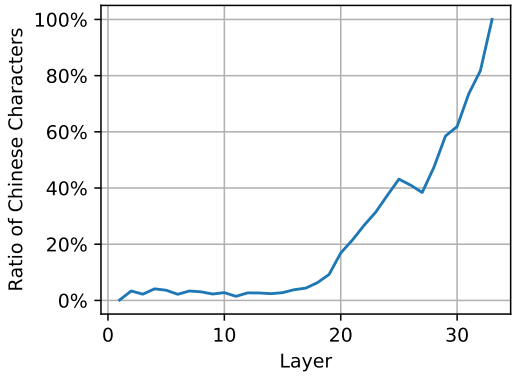

Figure 2: The ratio of generating Chinese tokens for each layer of Mistral-7B on solving the MGSM task (Chinese part) with chain-of-thought.

2.3 LLM's Three-Phase Working Pattern

A theoretical explanation for this language mismatch could be attributed to LLM's three-phase working pattern (Figure 3). In the work of Wendler et al. (2024), it is discovered that for simple multilingual in context learning prompts the forward computation of LLMs can be divided into three phases: understanding the context, generating the concept for the next token, and converting the concept into target language tokens. Furthermore, during this process, the change in prediction entropy serves as a crucial indicator for each specific working phase.

Position	DE	FR	ES	RU	ZH	AVG
[4, 8)	36.0	35.2	37.6	35.2	34.0	35.6
[8, 12)	**40.0**	39.2	**43.2**	**38.0**	34.8	39.0
[12, 16)	38.0	**39.6**	40.8	37.6	**40.0**	**39.2**
[16, 20)	37.2	33.6	38.4	37.2	32.0	35.7
[20, 24)	34.4	33.6	38.8	35.2	34.4	35.3
[24, 28)	33.2	36.0	35.2	30.4	34.8	33.9

Table 1: We set different positions for layer-skipping during contrastive decoding and observe Mistral-7B's reasoning accuracy on MGSM dataset.

According to this analysis, the early exit approach skips the final language conversion phase, leading to a language mismatch between amateur logits and expert logits. Therefore, a promising approach to address this issue is to skip only partial middles layers and complete the computation within the top transformer blocks. To validate this

[2]We explain the detailed setting of Figure 2 in Appendix A.

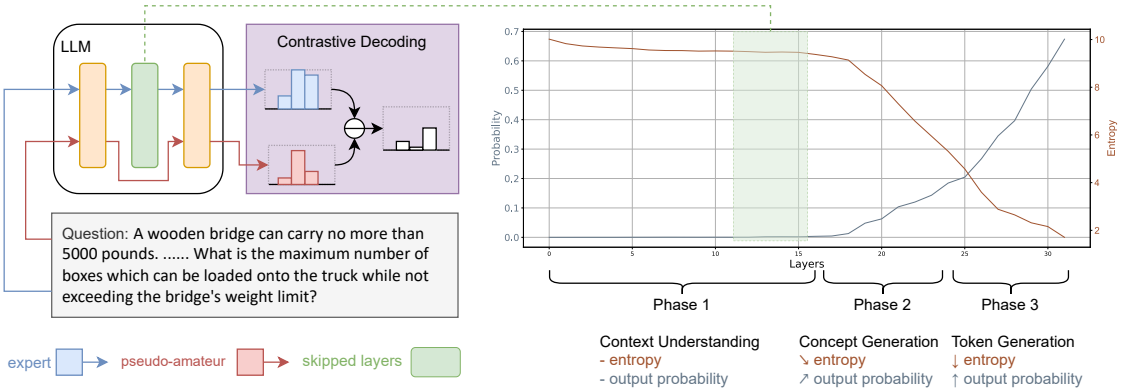

Figure 3: Illustration of our devised contrastive decoding approach. The idea of the line chart and three phases division are borrowed from the work of Wendler et al. (2024). In the line chart, "probability" denotes the token generation probability and "entropy" denotes the entropy of the prediction distribution.

idea, we conduct a preliminary study (Table 1) by skipping layers at different positions. The results indicate that skipping the layers in the lower half of the model (context understanding phase) produces more helpful amateur logits, whereas skipping the top layers most significantly degrades performance.

3 Methodology

Based on our previous discussion, we now introduce our devised algorithm in detail. Our intuition is that when the model's computation is perturbed during the context understanding phase, such as by skipping some layers, it tends to generate fluent but unreasonable content due to the poorly extracted features. This perturbation causes the output distribution to serve as useful amateur logits for contrastive decoding.

3.1 Overview

To obtain the amateur logits, our approach skips the layer span $[m, n)$ of the model during forward computation. After performing the layer skipping, the hidden state computation of a transformer model with N layers can be written as:

$$h_i = \begin{cases} \texttt{Emb}(x) & i = 0 \\ \texttt{L}_i(h_{i-1}) & i \in (0, m) \cup [n, N] \\ h_m & i \in [m, n) \end{cases}$$
$$p_a = f_{out}(h_N)$$

where x denotes a input token, h_i represents the output hidden states of the i-th layer and L_i represents the i-th layer of the model. The indices m and n mark the beginning and end of the layer skipping span. In the last layer, the prediction probability of amateur p_a can be obtained by grounding the hidden state h_N into the output embedding space via the output function f_{out}.

3.2 Strategies for Layer Skipping

Correctly setting the skipping span $[m, n)$ is non-trivial. We propose the following two strategies for position selection:

Heuristic layer skipping (SL-H) Considering that the first phase involves the bottom half of the layers in LLMs, a basic strategy is to randomly skip a few layers in this region, excluding the very bottom layers that process low-level lexical information. For each evaluated sample, we sample the begin-skipping layer m and set the end-skipping layer n with following equations:

$$\begin{aligned} m &= m \sim \mathcal{U}(4, N/2 - 1) \\ n &= m + \text{round}(N/8) \end{aligned} \tag{1}$$

Dynamic layer skipping (SL-D) To automatically determine the skipping position, we propose a dynamic algorithm based on entropy change. This approach is motivated by the sharp decrease in entropy that occurs between the first and second phase (Figure 3). Specifically, we calculate the entropy of the output distribution for each layer and identify the position where the entropy decreases by more than a predefined threshold δ. We set the end of the skipping span n to this position (Figure 3). The formal description of this algorithm is as follows:

$$\begin{aligned} m &= n - \text{round}(N/8) \\ n &= \min_{i \in \{k, \ldots, N\}} \{i : e'_i - e'_{i-1} > \delta\} \end{aligned} \tag{2}$$

	EN	DE	FR	ES	RU	ZH	JA	TH	TE	BN	SW	AVG	HRL	LRL
Mistral 7B														
Direct	45.6	33.2	34.4	34.8	32.0	34.4	19.2	16.4	2.0	12.0	6.0	24.5	33.4	9.1
DoLa	46.0	35.6	34.4	39.6	31.2	31.6	20.0	16.4	2.4	8.8	8.4	24.9 (+0.4)	34.1 (+0.7)	9.0 (-0.1)
SL-H	50.4	38.4	36.4	**42.0**	36.4	39.6	**24.8**	**19.6**	**6.8**	**16.0**	**10.8**	**29.2** (+4.7)	38.3 (+4.9)	**13.3** (+4.2)
SL-D	**54.0**	**38.8**	**38.8**	39.2	**40.4**	**41.2**	22.4	17.2	2.4	14.8	6.4	28.7 (+4.1)	**39.3** (+5.9)	10.2 (+1.1)
Deepseek 7B														
Direct	17.2	13.2	**16.4**	**18.0**	12.0	16.8	10.0	2.4	2.0	2.8	2.0	10.3	14.8	2.3
DoLa	11.2	6.0	7.6	6.4	8.0	14.4	2.4	1.6	2.0	1.6	2.4	5.8 (-4.5)	8.0 (-6.8)	1.9 (-0.4)
SL-H	20.8	17.2	12.8	14.0	14.8	**24.0**	10.4	**6.4**	2.4	**3.2**	**4.0**	11.8 (+1.6)	16.3 (+1.5)	**4.0** (+1.7)
SL-D	**24.0**	**18.0**	13.2	12.8	**15.2**	22.8	**12.4**	5.6	**2.8**	2.4	2.4	**12.0** (+1.7)	**16.9** (+2.1)	3.3 (+1.0)
Baichuan 2 7B														
Direct	29.6	14.8	**23.6**	20.0	16.8	27.2	10.4	8.8	1.6	4.0	3.6	14.6	20.3	4.5
DoLa	27.2	**19.2**	22.0	17.6	16.0	31.6	13.2	8.4	0.8	4.0	2.4	14.8 (+0.2)	21.0 (+0.6)	3.9 (-0.6)
SL-H	29.6	18.0	22.0	23.2	15.2	32.8	**13.6**	**10.8**	**2.4**	4.8	**4.8**	**16.1** (+1.5)	**22.1** (+1.7)	**5.7** (+1.2)
SL-D	**30.0**	14.8	18.4	**24.8**	**19.6**	**34.4**	11.6	9.6	**2.4**	**5.2**	3.6	15.9 (+1.3)	21.9 (+1.6)	5.2 (+0.7)
LLaMA 3 8B														
Direct	57.6	41.2	40.8	48.8	47.2	42.4	31.2	**42.0**	18.0	30.0	24.0	38.5	44.2	28.5
DoLa	56.8	42.0	41.6	49.2	43.6	41.6	31.2	38.4	19.6	26.8	26.4	37.9 (-0.5)	43.7 (-0.5)	27.8 (-0.7)
SL-H	**59.2**	**48.4**	42.0	**55.2**	**48.4**	43.6	**36.0**	40.8	**26.8**	**40.8**	28.0	**42.7** (+4.2)	**47.5** (+3.4)	**34.1** (+5.6)
SL-D	54.4	42.4	**43.6**	50.4	46.0	**45.6**	31.2	41.6	25.6	38.4	**28.4**	40.7 (+2.2)	44.8 (+0.6)	33.5 (+5.0)
LLaMA 2 13B														
Direct	34.8	21.6	22.0	26.0	20.0	20.4	13.6	6.4	1.2	1.6	2.8	15.5	22.6	3.0
DoLa	32.8	25.2	**25.6**	25.6	18.4	19.6	10.8	6.4	0.4	2.8	5.2	15.7 (+0.2)	22.6 (-0.1)	3.7 (+0.7)
SL-H	**36.4**	**27.6**	24.8	26.4	**22.0**	23.2	12.0	**7.6**	1.2	**5.2**	**6.0**	17.5 (+2.0)	24.6 (+2.0)	**5.0** (+2.0)
SL-D	34.8	26.0	24.4	**30.4**	21.6	**25.2**	**17.6**	4.8	**1.6**	4.4	3.6	**17.7** (+2.2)	**25.7** (+3.1)	3.6 (+0.6)

Table 2: Comparison results of different decoding methods on MGSM. "HRL" and "LRL" denote the average performance on seven high-resource languages and four low-resource languages. "SL-H" and "SL-D" denote the heuristic skipping and dynamic skipping of our approach. The bold and underlined text denotes the best and second best results along the column.

where e_i is the entropy of output probability of i-th layer computed by $-\sum f_{out}(h_i) \log f_{out}(h_i)$, and k is the minimum index that ensures $m > 6$, for excluding early layers[3].

4 Experiments

4.1 Settings

Expert LLMs In our experiments, we consider several popular LLMs with with different model sizes, including Mistral-7B (Jiang et al., 2023), Baichuan2-7B (Baichuan, 2023), Deepseek-7B (DeepSeek-AI, 2024), LLaMA3-8B (Meta, 2024) and LLaMA2-13B (Touvron et al., 2023b). During decoding, we use few-shot chain-of-thought prompting. More details are reported in Appendix B.

Baseline decoding algorithms We include three key baselines for comparison:

1. **Direct**: direct inference without using contrastive decoding.
2. **DoLa** (Chuang et al., 2023): the latest amateur-free contrastive decoding variant that uses early exit instead of completing the computation within the top transformer blocks.
3. **Vanilla** (Li et al., 2023): vanilla contrastive decoding approach that requires both expert model and an extra amateur model[4].

Evaluation datasets We consider both multilingual reasoning tasks MGSM[5] (Shi et al., 2022) and English reasoning benchmarks AQUA (Ling et al., 2017) , GSM8K (Cobbe et al., 2021) and GSM-PLUS (Li et al., 2024)[6] to evaluate the effectiveness of our proposed method. For all datasets, we report the accuracy as a performance metric.

4.2 Results

[3]In practical applications, to stabilize spikes or fluctuations in the entropy value e_i, we implement average pooling by computing $e_i' = (e_{i-1} + e_i)/2$. Additionally, we enforce a descending constraint such that $\forall j > n, e_j < e_n$.

[4]Note that only for Baichuan2-7B and LLaMA2-13B, we can find suitable amateur models: Baichuan's 220B token checkpoint and Sheared-LLaMA-1.3B from Xia et al. (2023) respectively, to implement vanilla contrastive decoding.

[5]The MGSM dataset contains English, six other high-resource languages and four low-resource languages.

[6]We evaluate on the subset of GSM-Plus where the answer is a integer.

	AQuA	GSM8K	GSM-Plus	AVG
		Mistral 7B		
Direct	32.3	43.6	34.7	36.9
DoLa	29.1	46.6	35.0	36.9 (+0.0)
SL-H	35.8	49.0	36.9	40.6 (+3.7)
SL-D	**37.0**	**50.6**	**37.4**	**41.6** (+4.8)
		Deepseek 7B		
Direct	24.4	15.0	11.9	17.1
DoLa	14.6	10.2	7.1	10.6 (-6.5)
SL-H	26.4	18.9	**13.7**	19.6 (+2.5)
SL-D	**28.7**	**20.3**	13.2	**20.8** (+3.6)
		Baichuan 2 7B		
Direct	25.2	20.3	16.0	20.5
DoLa	28.0	22.0	15.4	21.8 (+1.3)
SL-H	28.0	23.3	17.8	23.0 (+2.5)
SL-D	**29.5**	**23.6**	**18.0**	**23.7** (+3.2)
Vanilla	26.0	20.9	15.8	20.9 (+0.4)
		LLaMA 3 8B		
Direct	34.6	55.4	**43.9**	44.6
DoLa	37.4	49.6	37.0	41.3 (-3.3)
SL-H	**43.3**	53.3	41.1	**45.9** (+1.3)
SL-D	38.2	**56.0**	40.7	44.9 (+0.3)
		LLaMA 2 13B		
Direct	25.6	26.0	20.8	24.1
DoLa	29.1	27.7	19.5	25.5 (+1.3)
SL-H	28.3	28.0	21.7	26.0 (+1.9)
SL-D	**31.5**	30.5	22.1	28.0 (+3.9)
Vanilla	29.9	**32.1**	**23.4**	**28.5** (+4.4)

Table 3: Comparison results of different decoding method on English reasoning benchmarks.

Layer skipping provides helpful distribution for contrastive decoding In both Table 2 and Table 3, we can see that our approach (SL-H & SL-D) outperforms direct inference by a large margin in average. The dynamic layer skipping approach achieves better performance than heuristic layer skipping in most of benchmark results. These results demonstrate that our layer skipping approach provides helpful amateur logits for contrastive decoding.

Our devised approach enjoys more superiority in multilingual scenarios As shown in Table 2, our proposed approach improves the reasoning accuracy of all evaluated LLMs over baseline decoding algorithms. These results, especially the failure of DoLa in multilingual tasks demonstrates the importance of keeping top transformer layers during contrastive decoding.

To further illustrate the shortcomings in the design of DoLa, we present an ablation study in which we skip the computation both in the selected region [m, n) and in the remaining layers [n, N]. Experimental results in Appendix D show that maintaining the computation in the top layers is essential.

Our proposed approach eliminates the need for finding an extra amateur model for contrastive decoding Compared to the vanilla contrastive decoding approach, our proposed approach does not require an extra amateur model while achieving comparable performance (Table 3). This makes it more applicable in practical scenarios.

5 Conclusion

This paper is motivated by the observation of the failure of contrastive decoding variant DoLa on multilingual generation. Through empirical analysis and drawing inspiration from previous interpretability study on LLM's three-phase working pattern, we find that the failure stems from a language mismatch between the early exit output and the final output. To address this issue, we propose an improved contrastive decoding algorithm by skipping a set of lower layers and preserving the computation in the upper transformer blocks, which are essential for language transition. Experimental results on both multilingual and monolingual benchmarks demonstrate the effectiveness of our method.

Limitations

Below we discuss potential limitations of our work:

1. Extra inference cost: although contrastive decoding algorithms improve the generation quality of LLMs, they introduces additional computational cost for obtaining amateur logits. This results in a slower inference speed.
2. Limited model range: we conducted experiments using several popular LLMs, but the range of considered models may still be limited. For instance, we have not considered LLMs with the Mixture-of-Experts architecture.

Acknowledgement

Shujian Huang is the corresponding author. This work is supported by National Science Foundation of China (No. 62376116, 62176120). Wenhao Zhu is also supported by China Scholarship Council (No.202306190172).

References

Baichuan. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. 2023. Dola: Decoding by contrasting layers improves factuality in large language models. *arXiv preprint arXiv:2309.03883*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

DeepSeek-AI. 2024. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Qintong Li, Leyang Cui, Xueliang Zhao, Lingpeng Kong, and Wei Bi. 2024. Gsm-plus: A comprehensive benchmark for evaluating the robustness of llms as mathematical problem solvers. *arXiv preprint arXiv:2402.19255*.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12286–12312, Toronto, Canada. Association for Computational Linguistics.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167, Vancouver, Canada. Association for Computational Linguistics.

AI Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. *Meta AI*.

nostalgebraist. 2020. interpreting gpt: the logit lens. *LessWrong*.

Sean O'Brien and Mike Lewis. 2023. Contrastive decoding improves reasoning in large language models. *arXiv preprint arXiv:2309.09117*.

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

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023a. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in english? on the latent language of multilingual transformers. *arXiv preprint arXiv:2402.10588*.

Mengzhou Xia, Tianyu Gao, Zhiyuan Zeng, and Danqi Chen. 2023. Sheared llama: Accelerating language model pre-training via structured pruning. *arXiv preprint arXiv:2310.06694*.

Zheng Zhao, Emilio Monti, Jens Lehmann, and Haytham Assem. 2024. Enhancing contextual understanding in large language models through contrastive decoding. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*.

A Details of Preliminary Analysis

For the preliminary analysis in Section 2.2, we evaluate the Mistral 7B model on the MGSM dataset in Chinese. During generation, we apply the early exit approach to each layer to obtain the output probabilities for the vocabulary and observe the top-1 token from each layer. Subsequently, we examine whether each token from each layer is a Chinese character and compute the ratio of such tokens over all generation positions where the output token of the final layer is a Chinese character.

B Detailed Experiment Setting

We use the multilingual examples from Shi et al. (2022) to evaluate on the multilingual MGSM dataset. Specifically, we use 2-shot examples for Telugu, 4-shot examples for Bengali and Thai, and 8-shot examples for the remaining languages. For other datasets, we use the English examples from Wei et al. (2022), 4-shot for AQuA, 8-shot for GSM8K and GSM-Plus.

We use the modified contrastive decoding from O'Brien and Lewis (2023) for both of our approaches and the vanilla contrastive decoding with $\alpha = 0.1, \beta = 0.5$. We set $\delta = 0.1$ for the dynamic layer skipping. We use greedy decoding for all evaluated methods. Regarding the implementation of DoLa, we follow the implementation of the original paper (Chuang et al., 2023) and select the early exit layer in the lower half layers.

C Dataset Statistics

We use the development set of AQuA and test set of other datasets for evaluation. The dataset statistics are reported in Table 4.

Dataset	# Lang	# Sample
AQuA	1	254
GSM8K	1	1,319
GSM-Plus (subset)	1	8,651
mGSM	11	2,750

Table 4: Dataset statistics of our used datasets.

D Ablation Study

To further illustrate the shortcomings in the design of DoLa, we present an ablation study in which we skip the computation both in the selected region [m, n) and in the remaining layers [n, N]. The modified versions are called SL-H (E) and SL-D (E) in Table 5 . Experiment results show that SL-H/D (E) performs worse than our proposed method (SL-H/D), demonstrating the necessity of maintaining computations in the top layers.

E Used Scientific Artifacts

Below lists scientific artifacts that are used in our work. For the sake of ethic, our use of these artifacts is consistent with their intended use.

1. *Transformers (Apache-2.0 license)*, a framework that provides thousands of pretrained models to perform tasks on different modalities such as text, vision, and audio.

	EN	DE	FR	ES	RU	ZH	JA	TH	TE	BN	SW	AVG
Direct	45.6	33.2	34.4	34.8	32.0	34.4	19.2	16.4	2.0	12.0	6.0	24.5
DoLa	46.0	35.6	34.4	39.6	31.2	31.6	20.0	16.4	2.4	8.8	8.4	24.9 (+0.4)
SL-H	50.4	38.4	36.4	**42.0**	36.4	39.6	**24.8**	**19.6**	**6.8**	**16.0**	**10.8**	**29.2** (+4.7)
SL-H (E)	46.4	33.2	37.6	34.0	30.8	33.2	17.2	14.0	1.6	12.0	7.2	26.7 (+2.2)
SL-D	**54.0**	**38.8**	**38.8**	39.2	**40.4**	**41.2**	22.4	17.2	2.4	14.8	6.4	28.7 (+4.1)
SL-D (E)	45.2	30.8	34.0	34.4	27.6	30.4	18.8	14.0	0.8	12.4	4.0	22.9 (-2.6)

Table 5: Accuracy of early exit variant on MGSM with Mistral 7B.