Rethinking Layer Redundancy in Large Language Models: Calibration Objectives and Search for Depth Pruning

Anonymous Authors[1]

Abstract

Depth pruning improves the inference efficiency of large language models by removing Transformer blocks. Prior work has largely treated layer redundancy as an inherent structural property of pretrained networks, emphasizing importance criteria and search algorithms for identifying removable layers. In contrast, we adopt a *functional perspective*, where redundancy depends jointly on the model and the calibration objective, suggesting that a universal layer ranking may not exist. Through an empirical study across three LLM families, two calibration objectives, and seven search algorithms, we find that different objectives produce qualitatively different pruning patterns, while perplexity and downstream reasoning accuracy rankings often fail to align. In contrast, under a fixed objective, different search algorithms tend to converge to similar pruning solutions. Overall, our results suggest that the calibration objective may play a larger role than the particular search algorithm in determining which layers appear redundant.

1. Introduction

Large Language Models (LLMs) have achieved strong capabilities, but their scale incurs substantial deployment cost (Grattafiori et al., 2024; Yang et al., 2025). Among structured compression approaches, *depth pruning* removes entire Transformer blocks, reducing inference cost proportionally to the number of pruned layers (Men et al., 2025; Song et al., 2024). Prior work advances depth pruning along two axes: importance criteria (Men et al., 2025; Zhang et al., 2026; Wee et al., 2025) and search algorithms (Song et al., 2024; Chen et al., 2025; Tang et al., 2025; Sieberling et al., 2025; Jansen et al., 2026). Several works implicitly treat layer redundancy as an intrinsic property of the pretrained network. For example, ShortGPT (Men et al., 2025) ranks layers by cosine similarity independent of downstream tasks, while similarity- and magnitude-based methods (Song et al., 2024; Chen et al., 2025) likewise apply static rankings across objectives. We refer to this assumption as the *structural view* of redundancy.

We examine whether this view is consistent with the empirical behavior of LLMs under pruning. Our results suggest that redundancy is not an invariant property of the pretrained model alone, but depends jointly on the model and the calibration objective. In particular, a layer identified as redundant under language modeling perplexity may remain important for downstream reasoning, suggesting an *objective-conditioned* notion of redundancy.

We study this question by formulating depth pruning as the subset selection problem

$$S^* = \arg\min_{|S|=k} \mathcal{L}\big(\mathcal{D};\, f_{\theta \odot m_S}\big), \tag{1}$$

and disentangling the effects of calibration objectives and search algorithms. Prior work often varies both simultaneously, making it difficult to determine which component drives the resulting pruning behavior (Zhang et al., 2026; Jansen et al., 2026). We therefore compare seven search algorithms under two calibration objectives across three LLM families: language modeling perplexity on C4 (Raffel et al., 2020), and a task likelihood margin (Wee et al., 2025) on commonsense reasoning (Hu et al., 2023).

Our experiments reveal three consistent findings. First, pruning patterns differ substantially across objectives: perplexity-based pruning concentrates on contiguous mid-to-late layers, whereas task likelihood margin pruning produces more distributed removal patterns. Second, perplexity rank and downstream reasoning accuracy rank often disagree, with Spearman correlations ranging from strongly negative to near zero. Third, under a fixed objective, different search algorithms converge to similar pruning solutions with relatively small performance differences. Together, these results suggest that the calibration objective may play a larger role than the particular search algorithm in determining which layers appear redundant.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

Our contributions are threefold. We introduce and empirically examine a *functional view* of layer redundancy in LLM depth pruning. We disentangle calibration objectives and search algorithms across three LLM families, two objectives, and seven search algorithms in a controlled setting. Finally, we identify a systematic misalignment between calibration perplexity and downstream reasoning accuracy as an empirical consequence of objective-dependent redundancy.

2. Related Works

Pruning paradigms. Unstructured pruning (Sun et al., 2024; Frantar & Alistarh, 2023; Yun, 2024) achieves high sparsity but often requires specialized hardware for speedup. In contrast, structured pruning removes architectural components, enabling immediate efficiency gains on standard hardware. Among these, *depth pruning* removes entire Transformer blocks (Song et al., 2024), with inference cost scaling approximately with the number of retained layers.

Importance criteria. Prior work designs importance criteria to identify redundant layers, including cosine similarity (Men et al., 2025), inter-layer output similarity (Song et al., 2024), mutual information (Zhang et al., 2026), and prompt-conditioned routing (Wee et al., 2025). These methods assume that layer importance is an intrinsic property of the pretrained model and can be captured by a suitable criterion. We revisit this assumption and examine whether redundancy instead depends on the evaluation objective.

Search strategies and the metric-search confound. Most frameworks combine an importance metric with a search procedure. One-shot (Men et al., 2025) and greedy iterative (Song et al., 2024) methods make local decisions, motivating global approaches such as evolutionary search (Sieberling et al., 2025) and constrained binary optimization (Jansen et al., 2026) for the NP-hard subset selection problem (Natarajan, 1995). However, these methods often introduce both a new metric and a new search strategy, making improvements difficult to attribute. We address this by disentangling the two axes, holding one fixed while varying the other; the prior-guided GA and BO variants are used as controlled search procedures rather than standalone contributions.

3. Depth Pruning as Subset Selection

3.1. Problem formulation

Let f_{θ_0} denote a pretrained LLM with N Transformer blocks, indexed by $I \triangleq \{1, \dots, N\}$. We assume access to a calibration dataset $\mathcal{D}$ and an evaluation loss $\mathcal{L}(\mathcal{D}; \theta)$. Throughout this work, $\mathcal{L}$ is instantiated as either the negative log-likelihood or a task likelihood margin loss (Wee et al., 2025). A pruning decision is represented by a subset $S \subseteq I$ of removed layers, with induced mask $(m_S)_i \triangleq \mathbb{1}[i \notin S]$. We denote by $f_{\theta_0 \odot m_S}$ the resulting pruned model.

Given a pruning budget k, depth pruning seeks a subset $S \subseteq I$ with $|S| = k$ that minimizes degradation on $\mathcal{D}$:

$$S^\star \;=\; \arg\min_{\substack{S \subseteq I \\ |S|=k}} \mathcal{L}\big(\mathcal{D};\, \theta_0 \odot m_S\big). \tag{2}$$

Equation (2) highlights that depth pruning is fundamentally defined relative to a calibration objective $\mathcal{L}$ and dataset $\mathcal{D}$, rather than solely by properties of the pretrained model.

3.2. Objective-dependent redundancy

A key perspective of this work is that layer redundancy is not solely a property of the pretrained model θ_0, but of the tuple $(\theta_0, \mathcal{L}, \mathcal{D})$. Fixing θ_0 while varying $\mathcal{L}$ can, in principle, lead to different minimizers of equation (2), since distinct objectives may assign different importance to the same hidden-state perturbations.

We therefore adopt a relational definition: a layer $i \in I$ is *functionally redundant with respect to* $\mathcal{L}$ if $i \in S^\star$ for some optimal or near-optimal subset of equation (2). Under this definition, the *structural view* implicit in prior work corresponds to the assumption that the set of redundant layers is approximately invariant across objectives, such that a single universal ranking suffices. In contrast, the *functional view* considered here relaxes this assumption and treats redundancy as objective-dependent.

This perspective yields a direct empirical question: if redundancy is primarily structural, then changing the calibration objective should produce similar pruning patterns and preserve model rankings across evaluation criteria. Conversely, if redundancy is functional, then different objectives may induce different subsets of removable layers and favor different compressed models.

3.3. Why search alone may be insufficient

A further challenge is that the effect of removing multiple layers is generally non-additive. Define

$$\Delta(S) \triangleq \mathcal{L}(\mathcal{D}; \theta_0 \odot m_S) - \mathcal{L}(\mathcal{D}; \theta_0). \tag{3}$$

For disjoint subsets $S_1, S_2 \subseteq I$, in general,

$$\Delta(S_1 \cup S_2) \;\neq\; \Delta(S_1) + \Delta(S_2), \tag{4}$$

because removing one layer perturbs the hidden-state distribution encountered by subsequent layers, a phenomenon related to *reconstruction error explosion* (Huang et al., 2025).

As a result, local layer scores that estimate the effect of removing a single layer may not reliably predict jointly removable subsets. This motivates evaluating redundancy at

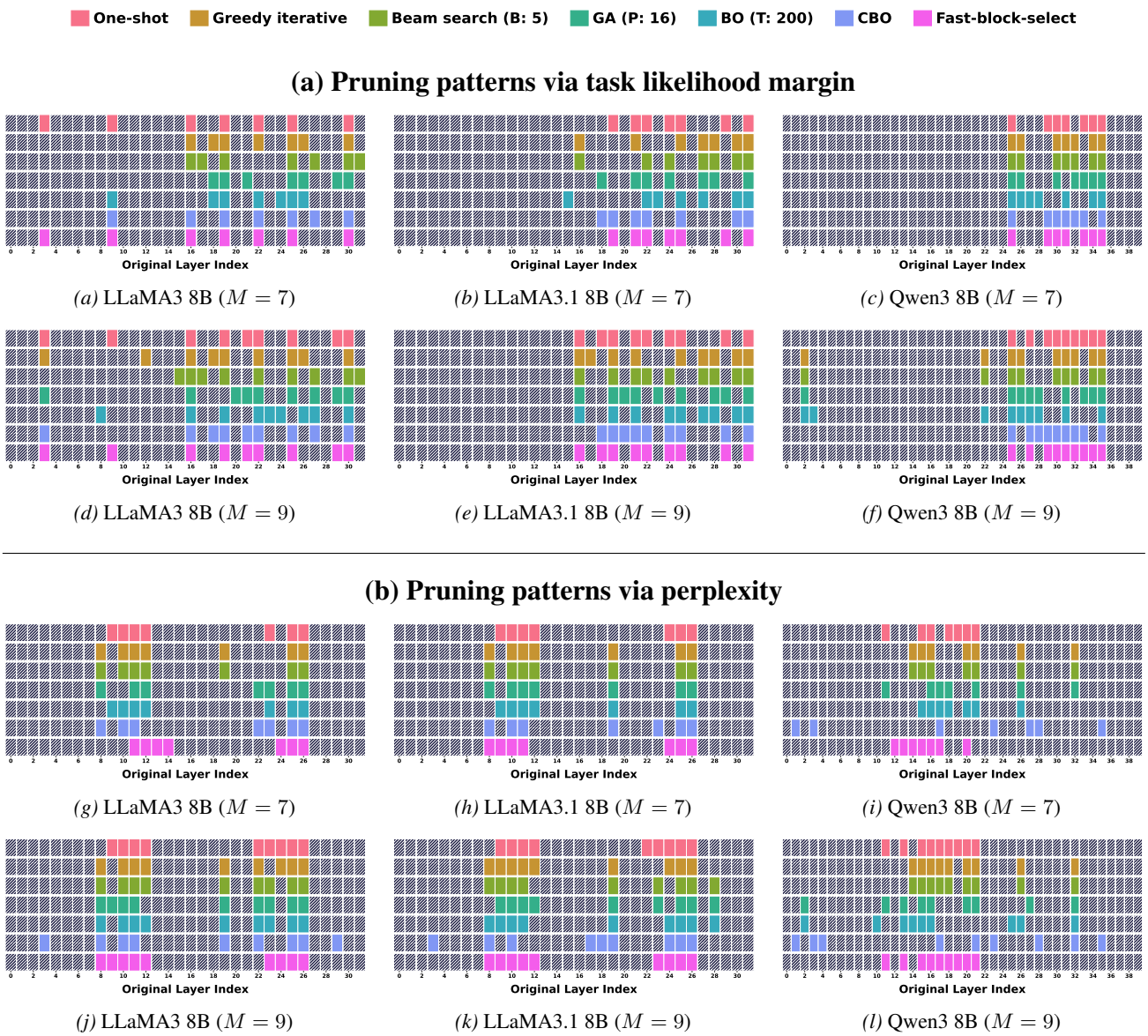


Figure 1. Pruning masks selected by each search algorithm across models and pruning scales ($M \in \{7, 9\}$). Top: task likelihood margin (calibration data: Commonsense 170k). Bottom: perplexity (calibration data: C4). Each row corresponds to a search algorithm; colored cells denote pruned layers.

the subset level rather than through independent layer scores. It also raises the question of whether changing the calibration objective alters pruning solutions more substantially than changing the search procedure itself.

4. Experimental Setup

For each pruned model, we vary the search algorithm while fixing the calibration objective, and conversely vary the objective while fixing the search procedure, isolating the two sources of variation in $S^\star$. All datasets are publicly available NLP benchmarks. We evaluate three open-weight LLMs of similar scale but different training pipelines: LLaMA3 8B, LLaMA3.1 8B, and Qwen3 8B.

Calibration objectives. We consider two losses $\mathcal{L}$ in equation (2): language modeling *perplexity* on C4 (Raffel et al., 2020) and a *task likelihood margin* loss on Commonsense 170k (Hu et al., 2023). Perplexity provides a task-agnostic fluency signal, while the margin is a discriminative, task-specific signal. If redundancy were primarily structural, the two objectives would be expected to favor similar layer subsets.

Search algorithms. Under each objective, we compare

Table 1. Zero-shot accuracy (↑) of task-likelihood-pruned models on downstream reasoning tasks for LLaMA3/3.1 and Qwen3 (8B). Each model is pruned by minimizing the task likelihood margin loss on Commonsense 170k, then evaluated without fine-tuning on HellaSwag (Hella), WinoGrande (Wino), ARC-Easy (ARC-E), ARC-Challenge (ARC-C), PIQA, and BoolQ. The Margin column reports the task likelihood margin (more negative is better). M denotes the number of removed layers. Best accuracy per pruning scale is in **bold**; best (most negative) Margin is underlined.

Model	M	Method	Hella	Wino	ARC-E	ARC-C	PIQA	BoolQ	Avg	Margin
LLaMA3 8B	0	Dense (Baseline)	79.23	73.72	77.53	54.10	80.63	82.26	74.58	-0.571
	7	One-shot	59.23	64.80	56.23	35.58	69.37	62.72	57.99	-0.312
		Greedy iterative	60.47	69.30	58.16	38.82	69.91	67.49	60.69	-0.408
		Beam search ($B = 5$)	61.04	64.17	54.17	39.59	70.73	71.04	60.12	-0.887
		GA ($P = 16$)	53.14	69.22	52.74	39.25	67.41	64.01	57.63	-0.106
		BO ($T = 200$)	64.34	67.32	60.06	36.95	71.98	64.77	**60.90**	-0.218
		CBO	59.63	66.54	54.00	36.35	69.97	73.09	59.93	-0.210
		Fast-block-select	59.23	64.80	56.23	35.58	69.37	62.72	57.99	-0.312
	9	One-shot	46.63	60.14	46.55	34.39	64.15	63.18	52.51	0.007
		Greedy iterative	53.28	63.46	49.87	31.66	65.13	63.21	54.43	-0.168
		Beam search ($B = 5$)	55.03	60.46	43.60	33.45	65.78	58.04	52.73	-0.271
		GA ($P = 16$)	49.65	63.38	48.32	34.98	64.74	65.87	**54.49**	-0.089
		BO ($T = 200$)	53.37	65.04	46.51	36.01	65.83	57.71	54.08	0.054
		CBO	51.86	62.75	47.01	34.56	62.73	64.22	53.85	0.006
		Fast-block-select	46.63	60.14	46.55	34.39	64.15	63.18	52.51	0.007
LLaMA3.1 8B	0	Dense (Baseline)	79.31	74.59	82.45	54.86	81.12	83.00	75.89	-0.615
	7	One-shot	63.62	66.61	58.84	43.00	69.04	73.67	**62.47**	-0.347
		Greedy iterative	60.19	64.48	59.39	40.27	70.51	71.04	60.98	-0.849
		Beam search ($B = 5$)	57.85	62.98	56.44	42.15	70.02	62.29	58.62	-0.652
		GA ($P = 16$)	64.01	63.61	54.25	40.10	70.78	63.94	59.45	-0.385
		BO ($T = 200$)	58.77	64.64	56.99	37.88	69.37	58.32	57.66	-0.543
		CBO	60.48	66.22	58.80	41.04	70.57	53.49	58.43	-0.469
		Fast-block-select	63.62	66.61	58.84	43.00	69.04	73.67	**62.47**	-0.347
	9	One-shot	54.81	62.67	49.83	38.31	65.02	66.94	56.26	-0.173
		Greedy iterative	55.94	63.69	50.17	36.43	68.44	64.83	56.58	-0.610
		Beam search ($B = 5$)	52.73	62.59	50.00	38.40	65.72	63.12	55.43	-0.546
		GA ($P = 16$)	50.26	64.09	50.63	38.57	66.21	52.08	53.64	0.033
		BO ($T = 200$)	52.20	63.46	52.36	39.08	66.43	63.67	56.20	-0.520
		CBO	55.04	62.12	49.66	37.71	67.19	68.29	**56.67**	-0.252
		Fast-block-select	54.81	62.67	49.83	38.31	65.02	66.94	56.26	-0.173
Qwen3 8B	0	Dense (Baseline)	74.94	67.72	80.93	56.66	77.69	86.57	74.09	-1.421
	7	One-shot	46.08	58.48	55.26	38.82	62.68	71.04	55.39	-1.630
		Greedy iterative	52.87	60.69	59.76	40.10	66.10	86.39	**60.99**	-2.410
		Beam search ($B = 5$)	52.87	60.69	59.76	40.10	66.10	86.39	**60.99**	-2.410
		GA ($P = 16$)	45.96	60.06	58.84	40.02	63.66	73.67	57.03	-1.863
		BO ($T = 200$)	54.52	58.01	60.90	38.40	65.67	81.50	59.83	-0.992
		CBO	52.80	63.30	58.25	40.36	65.34	83.85	60.65	-0.898
		Fast-block-select	46.08	58.48	55.26	38.82	62.68	71.04	55.39	-1.630
	9	One-shot	41.52	60.93	51.73	34.90	61.59	69.11	53.30	-2.536
		Greedy iterative	51.76	59.59	58.67	40.02	65.02	85.60	60.11	-1.915
		Beam search ($B = 5$)	51.76	59.59	58.67	40.02	65.02	85.60	60.11	-1.915
		GA ($P = 16$)	44.99	57.38	55.22	36.35	62.57	62.17	53.11	-1.127
		BO ($T = 200$)	57.16	60.30	61.45	39.16	66.81	78.84	**60.62**	-0.301
		CBO	47.29	59.51	46.38	34.90	62.68	79.11	54.98	-0.650
		Fast-block-select	41.52	60.93	51.73	34.90	61.59	69.11	53.30	-2.536

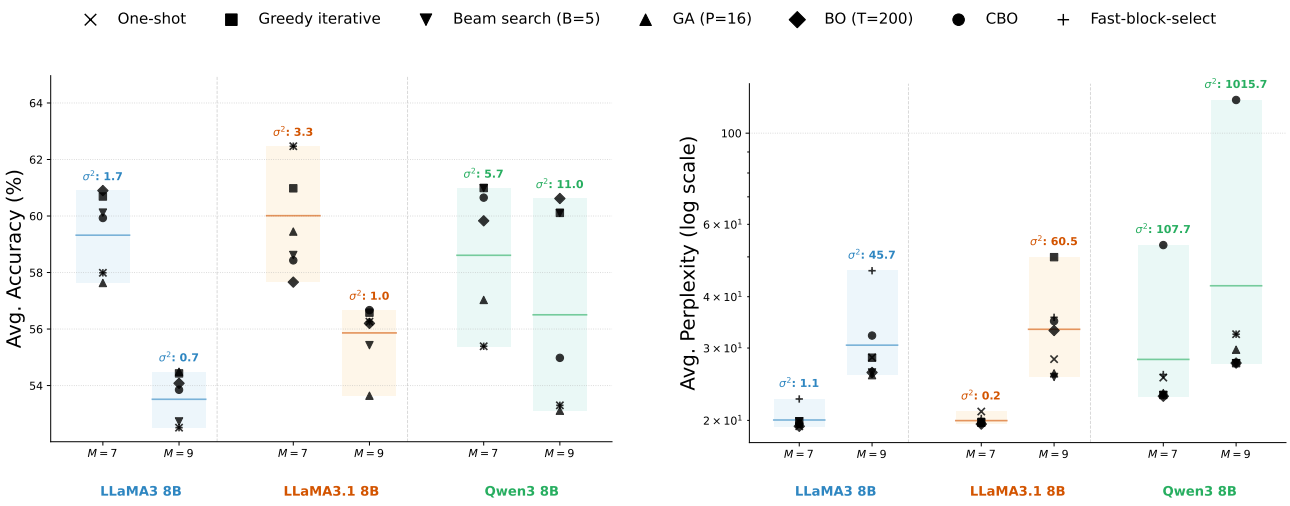
seven algorithms spanning the locality spectrum: one-shot top-k selection, greedy iterative deletion (Song et al., 2024), beam search ($B = 5$), prior-guided genetic algorithm (GA; population 16, elitism 0.2, mutation 0.15, 10 generations), prior-guided Bayesian optimization (BO; 200 trials, 10 random inits), constrained binary optimization (CBO) (Jansen et al., 2026), and fast-block-select (Zhang et al., 2026). These serve as a controlled panel of search behaviors rather than competing method proposals; all share the same $\mathcal{L}$ and $\mathcal{D}$ within each cell.

Pruning budgets. We examine two budgets $k \in 7, 9$, spanning moderate to high compression of the roughly 32–40 blocks in each model, from regimes where one-shot is typically considered adequate ($k = 7$) to regimes where reconstruction error becomes more pronounced ($k = 9$).

Evaluation. Evaluation is aligned with the calibration objective. Perplexity-pruned models are evaluated on WikiText-2 (Merity et al., 2017), C4 (Raffel et al., 2020), and LAMBADA (Paperno et al., 2016). Task-likelihood-pruned models are evaluated zero-shot on HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2019), ARC (Easy and Challenge) (Clark et al., 2018), PIQA (Bisk et al., 2020), and BoolQ.

Table 2. Perplexity (↓) of perplexity-pruned models across LLaMA3, LLaMA3.1, and Qwen3 (8B). Each model is pruned by minimizing calibration perplexity on C4, then evaluated on WikiText-2 (W2), C4, and LAMBADA (LMB). M denotes the number of removed layers. The best results for each pruning scale are highlighted in **bold**.

M	Method	LLaMA3 8B				LLaMA3.1 8B				Qwen3 8B			
		W2	C4	LMB	Avg.	W2	C4	LMB	Avg.	W2	C4	LMB	Avg.
0	Dense (Baseline)	6.14	8.31	16.11	10.18	6.24	8.37	16.21	10.27	9.71	13.65	23.56	15.64
7	One-shot	13.12	16.71	28.21	**19.35**	14.74	16.72	31.60	21.02	17.53	22.20	36.53	25.42
	Greedy iterative	13.12	16.70	29.88	19.90	13.10	16.58	29.64	19.77	15.55	19.92	33.70	23.06
	Beam ($B = 5$)	13.12	16.70	29.88	19.90	13.10	16.58	29.64	19.77	15.55	19.92	33.70	23.06
	GA ($P = 16$)	13.16	17.02	28.77	19.65	13.10	16.58	29.64	19.77	15.14	19.83	34.96	23.31
	BO ($T = 200$)	13.12	16.71	28.21	**19.35**	13.32	16.67	28.78	**19.59**	15.23	20.15	33.32	**22.90**
	CBO	13.33	17.21	28.30	19.61	13.30	16.85	28.89	19.68	28.72	34.74	96.82	53.42
	Fast-block-select	17.40	19.47	30.78	22.55	13.66	16.84	30.16	20.22	17.19	22.94	37.45	25.86
9	One-shot	21.47	23.25	40.65	28.46	20.85	22.71	41.02	28.19	24.77	29.51	42.97	32.42
	Greedy iterative	20.16	22.18	42.92	28.42	34.77	21.71	93.28	49.92	19.78	23.97	38.92	**27.56**
	Beam ($B = 5$)	18.27	21.98	38.22	26.16	17.88	21.28	37.35	**25.51**	19.78	23.97	38.92	**27.56**
	GA ($P = 16$)	18.31	22.61	36.38	**25.77**	18.77	21.49	37.84	26.03	21.52	24.24	43.41	29.72
	BO ($T = 200$)	18.27	21.98	38.22	26.16	22.42	21.33	55.56	33.10	19.15	23.62	39.96	27.58
	CBO	22.81	26.62	47.07	32.17	28.03	28.26	48.35	34.88	68.02	63.13	230.18	120.44
	Fast-block-select	45.43	23.57	69.81	46.27	29.91	22.36	54.62	35.63	24.77	29.51	42.97	32.42



(a) Zero-shot Accuracy (↑), task-likelihood-pruned models

(b) Perplexity (↓), perplexity-pruned models

Figure 2. Performance variance across search algorithms under same-metric evaluation. Each point represents a search method; shaded regions denote the range, horizontal lines the mean, and σ^2 the inter-method variance.

5. Functional View of Redundancy

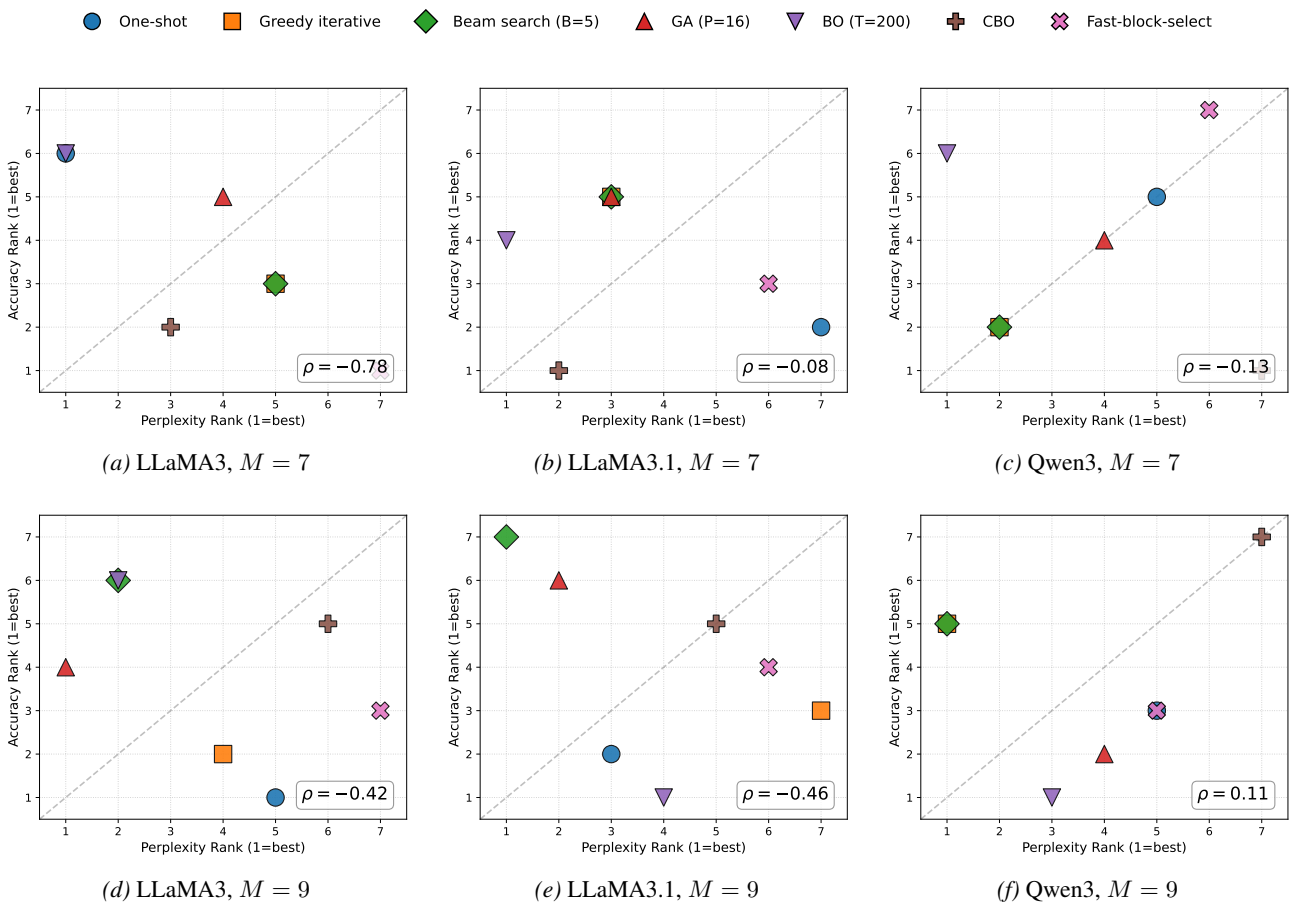
We present three empirical observations: (i) objectives change which layers are selected, (ii) perplexity and downstream accuracy rankings do not consistently align, and (iii) within a fixed objective, search algorithms yield similar solutions.

5.1. Objectives reshape pruning patterns

Figure 1 shows the removed layers under each objective. Under the structural view, both objectives should yield similar patterns. Instead, under *perplexity*, pruning concentrates in contiguous mid-to-late layer clusters that are consistent across algorithms. Under the *task likelihood margin*, removals are more distributed and vary across algorithms. The two objectives therefore induce qualitatively different redundancy patterns.

5.2. Perplexity vs. accuracy

We compare rankings of perplexity-pruned models by calibration perplexity and downstream accuracy. If redundancy were primarily structural, the rankings would be expected to align more closely. However, Spearman correlations (Figure 3) range from $\rho = -0.78$ to near zero, with no strong positive cases. The model that best preserves perplexity is often not the one that best preserves accuracy, indicating that the two objectives define different redundancy structures.



(a) LLaMA3, $M = 7$ *(b)* LLaMA3.1, $M = 7$ *(c)* Qwen3, $M = 7$

(d) LLaMA3, $M = 9$ *(e)* LLaMA3.1, $M = 9$ *(f)* Qwen3, $M = 9$

Figure 3. Perplexity rank vs. accuracy rank for perplexity-pruned models. Spearman ρ is negative or near-zero in most conditions, indicating that lower perplexity does not guarantee higher accuracy.

5.3. Limited variation across search algorithms

Within a fixed objective, variation across search algorithms is limited. For task-likelihood-pruned models (Figure 2a), inter-method variance is small ($\sigma^2 = 0.7$–5.7), and changing the search typically shifts accuracy by only 1–3 points. Perplexity-pruned models (Figure 2b) show similar behavior, except for a few outlier failures under certain methods.

In contrast, changing the objective while holding the search fixed produces substantially different layer selections and can alter model rankings. Thus, within-objective variation mainly perturbs performance, whereas changing the objective leads to substantially different pruning solutions.

The cost-benefit profile of search. Table 3 shows that more expensive search yields limited gains. On LLaMA3 8B at $M = 7$, one-shot completes in under 10 minutes at 57.99% accuracy, while BO takes about 70 minutes to reach 60.90%. The improvement remains below 3 points. These results suggest that the calibration objective may be a more influential factor than the search procedure in determining redundancy.

Table 3. Search time in seconds and average zero-shot accuracy (%) for LLaMA3 8B at M=7 under task likelihood margin pruning.

Method	Time (s)	Average acc. (%)
One-shot	550.52	57.99
Greedy iterative	3,222.51	60.69
Beam Search ($B = 5$)	13,869.36	60.12
GA ($P = 16$)	3,543.53	57.63
BO ($T = 200$)	4,233.40	60.90
CBO	8,172.51	59.93
Fast Block Select	428.54	57.99

6. Conclusion

We examine whether layer redundancy in LLM depth pruning is structural or jointly determined by the evaluation objective. By disentangling search and objective across models, objectives, and algorithms, we find evidence consistent with the *functional view*: objectives induce different pruning patterns, perplexity and accuracy rankings do not align, and search algorithms yield similar solutions under a fixed objective. These results suggest that the calibration objective may be more influential than the choice of search algorithm.

References

Bisk, Y., Zellers, R., Bras, R. L., Gao, J., and Choi, Y. Piqa: Reasoning about physical commonsense in natural language. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.

Chen, X., Zhang, H., Zeng, F., Wei, Y., Wang, Y., Ling, X., Li, G., and Yuan, C. Prune&comp: Free lunch for layer-pruned llms via iterative pruning with magnitude compensation. *arXiv preprint arXiv:2507.18212*, 2025.

Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., and Tafjord, O. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv*, abs/1803.05457, 2018.

Frantar, E. and Alistarh, D. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *ICML*, pp. 10323–10337, 2023. URL https://proceedings.mlr.press/v202/frantar23a.html.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Hu, Z., Wang, L., Lan, Y., Xu, W., Lim, E.-P., Bing, L., Xu, X., Poria, S., and Lee, R. LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5254–5276, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.319. URL https://aclanthology.org/2023.emnlp-main.319/.

Huang, W., Zhang, Y., Zheng, X., Chao, F., and Ji, R. Determining layer-wise sparsity for large language models through a theoretical perspective. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=otNB7BzsiR.

Jansen, D., Rausch, R., Montero, D., and Orus, R. Block removal for large language models through constrained binary optimization. *arXiv preprint arXiv:2602.00161*, 2026.

Men, X., Xu, M., Zhang, Q., Yuan, Q., Wang, B., Lin, H., Lu, Y., Han, X., and Chen, W. ShortGPT: Layers in large language models are more redundant than you expect. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20192–20204, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1035. URL https://aclanthology.org/2025.findings-acl.1035/.

Merity, S., Xiong, C., Bradbury, J., and Socher, R. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Byj72udxe.

Natarajan, B. K. Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24(2):227–234, 1995. doi: 10.1137/S0097539792240406. URL https://doi.org/10.1137/S0097539792240406.

Paperno, D., Kruszewski, G., Lazaridou, A., Pham, N. Q., Bernardi, R., Pezzelle, S., Baroni, M., Boleda, G., and Fernandez, R. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1525–1534, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P16-1144.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21 (140):1–67, 2020.

Sakaguchi, K., Bras, R. L., Bhagavatula, C., and Choi, Y. Winogrande: An adversarial winograd schema challenge at scale. *arXiv preprint arXiv:1907.10641*, 2019.

Sieberling, O., Kuznedelev, D., Kurtic, E., and Alistarh, D. Evopress: Accurate dynamic model compression via evolutionary search. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=l7QzcZpjc5.

Song, J., Oh, K., Kim, T., Kim, H., Kim, Y., and Kim, J.-J. Sleb: Streamlining llms through redundancy verification and elimination of transformer blocks. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

Sun, M., Liu, Z., Bair, A., and Kolter, J. Z. A simple and effective pruning approach for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=PxoFut3dWW.

Tang, S., Sieberling, O., Kurtic, E., Shen, Z., and Alistarh, D. Darwinlm: Evolutionary structured pruning of large language models. *arXiv preprint arXiv:2502.07780*, 2025.

Wee, J., Park, M., and Lee, J. Prompt-based depth pruning of large language models. In *International Conference on Machine Learning*, 2025.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Yun, J. Robust neural pruning with gradient sampling optimization for residual neural networks. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–10, 2024. doi: 10.1109/IJCNN60899.2024.10650301.

Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., and Choi, Y. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

Zhang, H., Zhang, Z., Wu, G., Chen, H., Guo, J., and Cheng, X. Mi-prun: Optimize large language model pruning via mutual information. *arXiv preprint arXiv:2601.07212*, 2026.