HARD TO REMOVE, CHEAP TO REPLACE: COMPACT, TRANSFERABLE SURROGATES FOR SENSITIVE LLM LAYERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Removing whole layers is a direct way to compress a large language model, but the damage it causes varies widely across layers and tasks, and removal alone does not show whether that damage can be recovered. We study layer sensitivity through recovery: how much of a removed layer's contribution a compact learned replacement can restore, and whether that replacement transfers to tasks it was not fitted on. Using Llama 3.1 8B and Qwen3-8B, with the rest of each model frozen, we apply three interventions to complete layers during free text generation: Skip removes a layer, Replace substitutes a small surrogate fitted to the layer's residual output, and Transfer reuses a surrogate fitted on a different dataset. We extend these interventions to pairs of layers, compare linear and nonlinear surrogates, and contrast trained surrogates with calibrated random maps. A linear surrogate of rank 64 with about 1/413 of a Llama layer's parameters improves over Skip in all 12 sensitive Llama settings, recovering up to 93.2% of the lost performance. Surrogates also transfer across datasets and task categories: in Llama, surrogates fitted in one task can transfer to other tasks, still with significant improvement over Skip. Jointly fitting two surrogates for two layers improves the replacement over independent surrogate composition. And we prove the effectiveness of the surrogates training by showing it outperforms random maps in all tasks.

1 INTRODUCTION

Reducing the parameter count of large language models while preserving their performance is a central goal of model compression. Removing entire Transformer layers offers a direct route to that goal, but its effect varies across layers and tasks (Gromov et al., 2025; Men et al., 2025; Hinostroza et al., 2026). When skipping a layer causes substantial damage, recovery becomes the key question: how much of the lost performance can a smaller replacement restore?

Studies of layer pruning use removal tests, similarity measures or learned skipping to identify computation that can be omitted (Men et al., 2025; Lawson & Aitchison, 2025). PoLar predicts programs tailored to each input that skip or repeat frozen pretrained layers (Li et al., 2026). We study sensitivity through recovery: pairing removal with a compact learned replacement measures what a layer costs to remove and how much of its behavior can be restored. We evaluate free text generation with interventions on complete layers during both prefill and decode.

Learned replacement is not new, which replaces parts of the original large model with surrogate modules. Intermediate teacher supervision dates back to FitNets (Romero et al., 2015), and LLM-Streamline, ReplaceMe and SubFit approximate removed blocks or submodules with fitted transformations (Chen et al., 2025; Shopkhoev et al., 2025; Cunegatti et al., 2026); ReplaceMe and SubFit also examine how calibration affects performance. LoRA adapts retained weights through updates of low rank (Hu et al., 2022), while GRASP compresses redundant layers by retaining selected singular components of their weight matrices (Liu et al., 2025). To further compress the model, our surrogate instead fits a single map from the incoming hidden state to the complete layer's residual. Prior work also connects linearity between adjacent states to pruning and evaluates fitted linear replacements (Razzhigaev et al., 2024). Recovery also depends on the task: Shrestha et al. (2026) find that generative reasoning can remain fragile under pruning even when classification

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

recovers. We examine how well a surrogate fitted on one dataset transfers to others and how this varies with depth. Full transfer grids across sources and targets measure how much layer behavior is shared across tasks, while comparisons of surrogate architectures and of replacements for layer pairs examine how recovery is achieved.

We study these questions in Llama 3.1 8B and Qwen3-8B using three interventions: Skip (remove a layer), Replace (substitute a small learned surrogate) and Transfer (reuse a surrogate fitted on another dataset), plus an ablation with random maps.

Our study makes three contributions:

- **We characterize sensitive layers through both removal and recovery.** Large Skip damage can coexist with strong recovery: a linear surrogate of rank 64 with about 1/413 of a layer's parameters recovers up to 93.2% of Skip damage across all 12 sensitive Llama settings.
- **We measure how much layer behavior is shared across tasks through transfer.** Frozen surrogates can recover performance across datasets and task categories, with effectiveness depending on the combination of source and target and the replaced layer.
- **We extend Skip, Replace, and Transfer to pairs of layers.** We study layer interactions and reuse across tasks; joint fitting improves replacement on the same dataset by up to 18.5 points over independent composition.

2 Intervention Framework

Our aim is to reduce the parameters used at selected layers while preserving model performance. A layer is a complete Transformer block, including attention and feedforward sublayers; indices start at zero. A *setting* combines one dataset with one layer; a *layer pair* contains two layers intervened on together. We ask two questions. **RQ1:** How much performance lost through layer removal can a small surrogate recover, how does recovery depend on its architecture, and does it hold when two layers are replaced together? **RQ2:** Can one surrogate or a pair of surrogates fitted on one dataset support performance on another without further training?

2.1 Removal and residual replacement

Let $h_\ell^{\text{dense}}(x,t) \in \mathbb{R}^d$ be the hidden representation entering layer ℓ of the original model for sequence x at token position t. The layer's residual output, which includes both sublayers' contributions, is

$$r_\ell(x,t) = h_{\ell+1}^{\text{dense}}(x,t) - h_\ell^{\text{dense}}(x,t). \tag{1}$$

A linear surrogate approximates it with an affine map of rank at most r,

$$g_{\ell,r}(h) = UV^\top h + b, \qquad \hat{h}_{\ell+1} = h_\ell + g_{\ell,r}(h_\ell), \tag{2}$$

with trainable U, V $\in \mathbb{R}^{d\times r}$ and b $\in \mathbb{R}^d$; both models have d = 4,096. For the nonlinear comparison, a bottleneck MLP with m SiLU units predicts the residual in the same way (Appendix A.1). Skip corresponds to g(h) = 0. Both interventions act throughout prefill and decode while all other layers remain in place. The surrogate operates on each token: it has no attention and keeps no KV cache for the replaced layer, relying on context already encoded by earlier layers.

With the pretrained model frozen, the surrogate parameters θ minimize the residual error over all token positions of the source training sequences $\mathcal{D}_S$, each a prompt followed by a fixed reference answer (Appendix C.1):

$$\mathcal{L}_{\ell,S}(\theta) = \frac{1}{N_S} \sum_{x\in\mathcal{D}_S} \sum_{t=1}^{T_x} \left\| g_{\ell,\theta}\big(h_\ell^{\text{dense}}(x,t)\big) - r_\ell(x,t) \right\|_2^2, \tag{3}$$

where T_x is the number of positions in sequence x and $N_S = \sum_x T_x$. Training uses dense states obtained by teacher forcing on reference answers, whereas at evaluation the surrogate receives states produced from the model's own generated tokens. Optimization details are in Appendix A.1.

2.2 MEASURING RECOVERY

On a shared evaluation set, let S_{dense}, S_{skip} and S_{rep} be the task scores of the original model, the model with the selected layer skipped and the model with that layer replaced. We define **Damage**, **Improvement** over Skip and **Recovery**, the fraction of Skip damage restored, as

$$D_{\text{skip}} = S_{\text{dense}} - S_{\text{skip}}, \qquad D_{\text{rep}} = S_{\text{dense}} - S_{\text{rep}}, \tag{4}$$

$$I = S_{\text{rep}} - S_{\text{skip}}, \qquad R = \frac{S_{\text{rep}} - S_{\text{skip}}}{S_{\text{dense}} - S_{\text{skip}}}. \tag{5}$$

R = 1 means replacement restores the Dense score, and R > 1 means it exceeds that score on the evaluated examples. Scores are on a 0–1 scale; we report differences in points ($\times 100$) and recovery as 100R%, and we report R only when $D_{skip} > 0.01$ (1 point).

2.3 FROZEN TRANSFER

For transfer, surrogates fitted on a source dataset replace the same layer or layer pair on a target dataset, with parameters and normalization fixed. We compare their score $S_{transfer}$ with S_{local}, the score of local surrogates fitted on the target dataset at the same layers, rank and fitting mode (independent or joint):

$$D_{\text{transfer}} = S_{\text{dense}} - S_{\text{transfer}}, \qquad G = S_{\text{local}} - S_{\text{transfer}}. \tag{6}$$

Together with improvement over Skip, $D_{transfer}$ and the gap G separate recovery from removal, retention of Dense performance and the value of fitting on the target; positive G favors local fitting. For a layer pair, recovery uses simultaneous Skip of both layers (Pair Skip) as its baseline.

3 EXPERIMENTAL DESIGN

We evaluate Skip, Replace and Transfer, alongside an ablation with random maps, in Llama 3.1 8B (Grattafiori et al., 2024) and Qwen3-8B with thinking disabled (Yang et al., 2025); the rest of the model stays frozen, and each intervention acts on complete layers throughout prefill and decode. We report every evaluated condition, including negative recovery, recovery above 100% and harmful replacements; full grids are in the appendix and supplementary material.

Datasets and capacity. The common suite contains ARC Easy (ARC; Clark et al., 2018), PIQA (Bisk et al., 2020), ASDiv (Miao et al., 2020), DART Math L5 (DART; Tong et al., 2024), ROPES (Lin et al., 2019) and SMS Spam (SMS; Almeida & Hidalgo, 2011); AG News (Zhang et al., 2015) and SQuAD v1 (SQuAD; Rajpurkar et al., 2016) provide additional Llama evaluations. We use the short names in parentheses throughout. The main replacement protocol uses 300 training, 100 validation and 100 test sequences per dataset. We write r32/r64/r128 for linear surrogates of rank r and MLP64 for a bottleneck MLP with 64 units. r64 is the primary configuration, and r32 serves as a capacity comparison for transfer.

Scoring and uncertainty. Greedy generation uses the same prompts for Dense, Skip and Replace within each comparison. ROPES and SQuAD use token F1 and the other tasks accuracy; scores are never pooled across tasks. For Replace and Transfer, we average scores from two training seeds within each test example before computing paired 95% bootstrap intervals. Here, *paired* means resampling the same test units across compared conditions. Appendix C.2 gives the generation, parsing and exploratory resampling procedures.

3.1 SKIP: REMOVAL SENSITIVITY

Scans of individual layers measure the change from Dense across 13 selected Llama layers on eight datasets and all 36 Qwen layers on six datasets, using 500 examples per task. Simultaneous Skip at four fixed layer pairs (Llama L2+L11 and L11+L31, Qwen L4+L17 and L4+L20) tests whether the effects of two removals are additive.

3.2 REPLACE: RECOVERY AND COMPOSITION

The Llama local study, in which each surrogate is trained and evaluated on the same dataset, covers two sensitive layers and one control which is not that sensitive to removal (i.e., with little removal

damage) per dataset (Figure 1); Qwen covers eight selected layers on all six datasets. Linear r64 and MLP64 have nearly identical parameter counts (about 528,000), enabling a comparison of function classes at similar capacity.

For the same four layer pairs used in Skip, *Independent* inserts two separately fitted surrogates, while *Joint* optimizes both together to match the dense model's representation after the second layer (Figure 2). Both modes replace the same two layers. Joint training starts from the independent weights; the primary comparison is Joint minus Independent on matched examples. Appendix C.3 gives the protocol for layer pairs.

3.3 Transfer: reuse across tasks

Each source surrogate is trained on one source dataset, then frozen and evaluated at the same layer on all six target datasets, forming full 6×6 grids at L2/L11/L31 in Llama and L4/L17/L35 in Qwen, at both ranks. Llama transfer also evaluates AG News and SQuAD, which are not used for surrogate training. For the four layer pairs used in Skip and Replace, both surrogates are fitted on the same source, independently or jointly, then transferred together to all six targets at both ranks.

3.4 Ablation with random maps

To examine the impact of the learned surrogates, at Llama L2/L11/L31 we compare one trained r64 surrogate with five *random maps* per dataset: surrogates of the same shape whose parameters are sampled at random and scaled on source validation data to match the root mean square of the original layer's residual outputs (Appendix B.2). We report the task score of learned surrogates minus the mean score of random surrogates, alongside Dense and Skip. Figure 5 shows the results of four representative tasks, while full results are in Table B2.

4 Results

4.1 Skip: sensitivity varies across tasks

Removal profiles reveal which layers each task depends on. Llama L31 is critical for question answering: removing it costs ROPES 73.0 F1 points and also sharply reduces SQuAD performance, while leaving AG News unchanged and improving SMS. Qwen shows a shared sensitivity at its input, with L0 producing the largest loss on all six tasks, followed by profiles that vary by task at later layers (Figure A3). These profiles identify sensitive layers where replacement can recover useful computation.

Joint removal of two layers reveals interactions that were missing in the individual removals. On ARC, removing Llama L2+L11 or Qwen L4+L17 together causes less damage than the sum of the individual removals, whereas Llama L11+L31 and Qwen L4+L20 cause more. A layer pair therefore needs to be evaluated directly: its combined effect cannot be inferred by adding two losses from individual removals.

4.2 Replace: small surrogates recover sensitive layers

4.2.1 Recovery with a reduced parameter budget

Linear r64 recovers performance in every sensitive Llama setting. Across all 12 settings, mean recovery ranges from 50.0% to 93.2%, using about 1/413 of the replaced layer's original parameters (Figure 1). Thus, even when removing a layer causes a large loss, a small fitted surrogate can restore much of the lost performance. Appendix A.4 gives all 12 r64 results and an r32 comparison from the same evaluation.

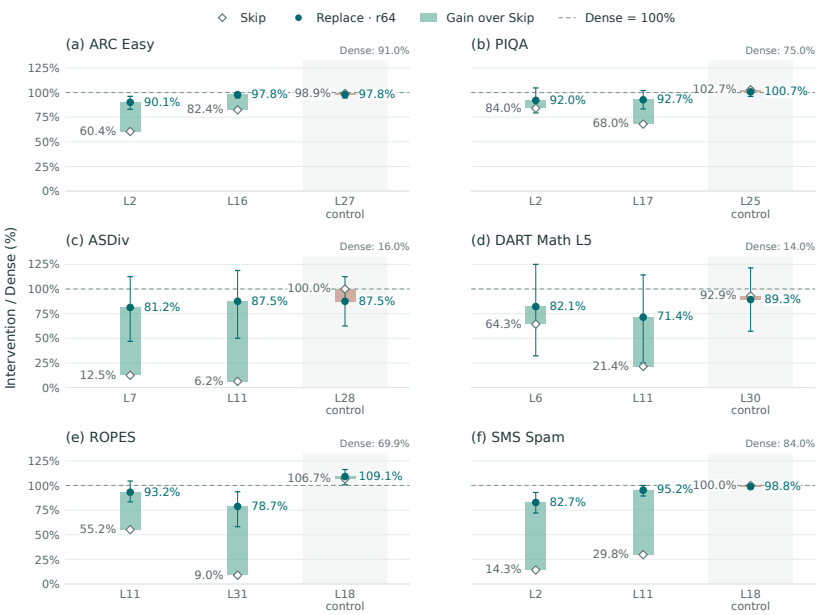


Figure 1: Small surrogates restore performance toward Dense. The vertical axis is $100 \times S_{\text{intervention}}/S_{\text{dense}}$, with each task's Dense score fixed at 100%. Gray diamonds mark Skip and teal circles mark linear r64 replacement; green bars span the gain over Skip, and muted rust marks decreases. Whiskers rescale the paired 95% damage intervals by the Dense score. The figure shows all 12 sensitive settings and six controls with little damage. Each comparison uses 100 matched examples. ROPES uses token F1 with 28 background groups; other tasks use accuracy. Linear r64 has 528,384 parameters, about 1/413 of a layer.

A larger surrogate does not necessarily yield better recovery. Increasing linear rank reduces residual reconstruction error across all 18 Llama settings, but task scores do not improve consistently with rank. This separates the ability to fit residuals from the ability to preserve task performance (more results in Appendix A.2).

4.2.2 Recovery changes with surrogate architecture

A nonlinear MLP recovers Qwen L6 where larger linear surrogates fail. MLP64 brings ARC and PIQA accuracy close to Dense with almost the same parameter count as linear r64, and outperforms linear r128 despite using roughly half as many parameters (Table 1). At this layer, changing the surrogate architecture is more effective than simply increasing linear rank, given the similar parameter counts between MLP and linear map.

Table 1: Qwen L6: MLP64 outperforms all tested linear ranks across six datasets. Scores use 100 test examples per task; ROPES uses token F1 and other tasks use the accuracy scorers in Appendix C.2. MLP64 has 528,448 parameters versus 528,384 for linear r64. Figure A5 shows all eight layers; paired intervals are in the supplementary material.

Task	Dense (%)	Skip (%)	Linear r32 (%)	Linear r64 (%)	Linear r128 (%)	MLP64 (%)
ARC	79.0	44.0	2.0	4.5	1.5	80.0
PIQA	78.0	33.0	19.5	13.0	10.5	78.5
ASDiv	54.0	15.0	0.0	0.5	0.0	37.0
DART	1.0	2.0	2.0	1.0	0.0	3.5
ROPES	81.4	14.4	17.9	39.1	37.1	73.4
SMS	72.0	68.0	6.0	8.0	7.5	41.0

MLP64's advantage over the tested linear surrogates spans all six tasks at Qwen L6, in some cases even outperforms Dense (Table 1). Together with the comparison at matched parameter counts, these results identify architecture as a useful choice for recovering this sensitive layer. Appendices A.2 and A.3 report the broader Llama and Qwen architecture comparisons.

4.2.3 JOINT TRAINING IMPROVES REPLACEMENTS FOR LAYER PAIRS

Joint training improves most evaluated replacements for layer pairs on the same dataset. Appendix B reports the results of replacing a pair of layers with independently trained VS jointly trained surrogates. Across four different layer pairs, joint fitting improves mean scores over independent composition in 30 of 48 combinations of task, layer pair and rank. The clearest benefit is on SMS at Llama L2+L11: joint r64 reaches Dense accuracy, and r32 gains 18.5 points over independent composition. Figure 2 shows these gains at Llama L2+L11, alongside single replacements and Pair Skip on the same test examples and layer pair, showing the value of fitting two surrogates together.

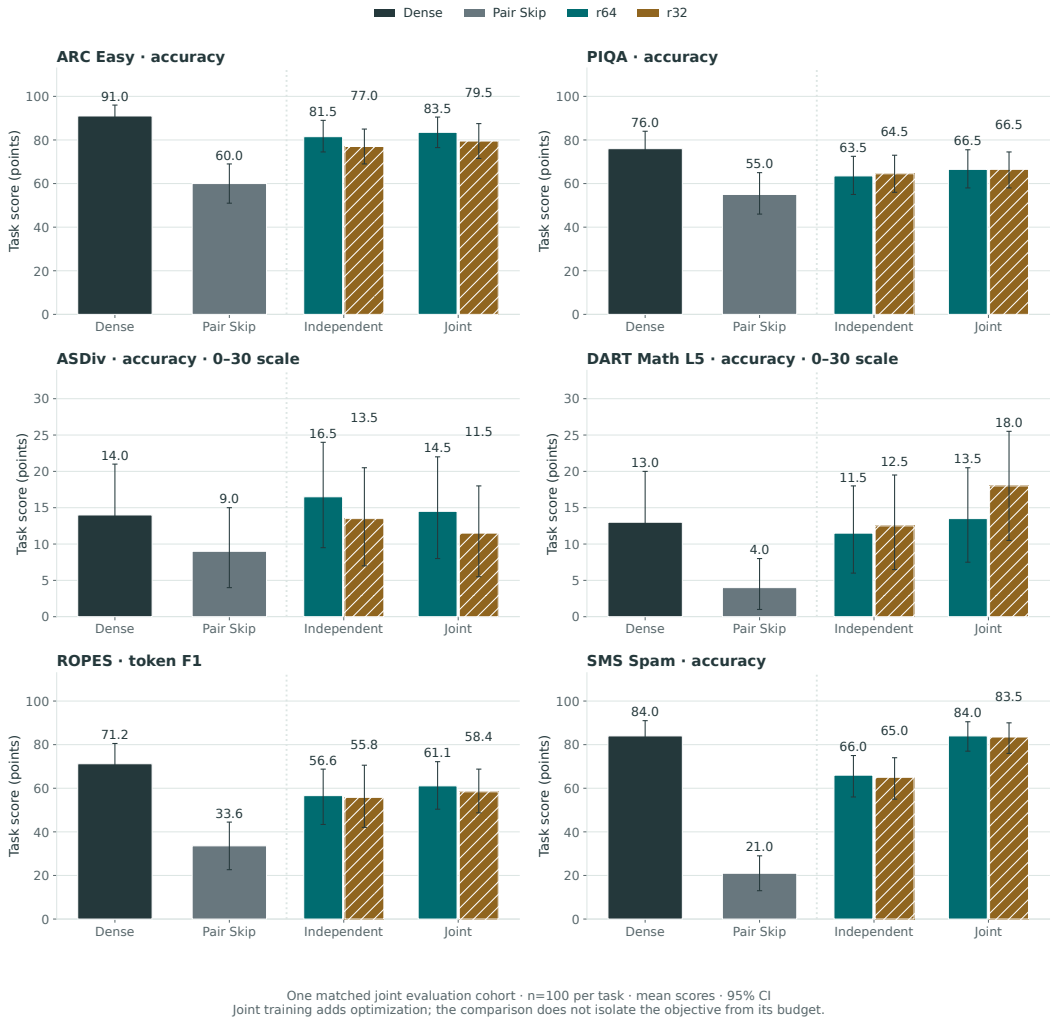


Figure 2: Independent and joint replacement at Llama L2 + L11. Dense, Pair Skip, independent composition and joint replacement are all evaluated on the same 100 examples per task, with paired 95% intervals (28 background groups for ROPES). Both ranks are shown. The axes span 0–30 points for ASDiv and DART and 0–100 for other tasks.

The benefit also appears in Qwen: at ASDiv L4+L20, joint fitting improves over independent composition at both ranks, with paired gain intervals above zero (Table B1). These examples show that fitting surrogates together can improve their combined task performance. Table B1 reports all 48 comparisons across tasks, layer pairs and ranks. In the future work, we will explore the potential of replacing more than two layers.

4.3 TRANSFER: REUSE DEPENDS ON TASK AND DEPTH

4.3.1 TRANSFER PERFORMANCE VARIES ACROSS TASKS AND LAYERS

Surrogates transfer across datasets, including between task categories. At Llama r64, transferred surrogates improve over Skip in 29 of 30 combinations with different sources and targets at L2 and 26 of 30 at L11. Each combination uses a surrogate fitted on the row's source dataset and evaluates it on the column's target dataset (Figure 3).

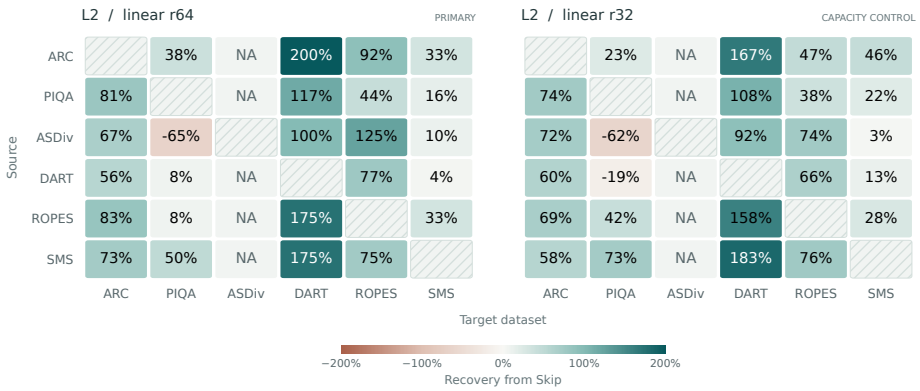


Figure 3(a): Llama transfer at L2. Cells show mean recovery of Skip damage (%) at r64 and r32; rows are training sources and columns are evaluation targets. Each target has 100 examples (28 backgrounds for ROPES). Hatched diagonals mark omitted cells with identical sources and targets; NA means Dense − Skip $\leq$ 1 point. ROPES uses token F1; other tasks use accuracy.

For example, a surrogate trained on ROPES at Llama L11/r64 exceeds both Dense and the local surrogate on ASDiv, recovering 146.2% of Skip damage. The effect of depth differs by target: for surrogates trained on ROPES, recovery of Skip damage is higher at L11 than L2 on SMS, but lower on ARC (Figure 3). Useful reuse of the replacement across tasks therefore depends on the source, target and layer together.

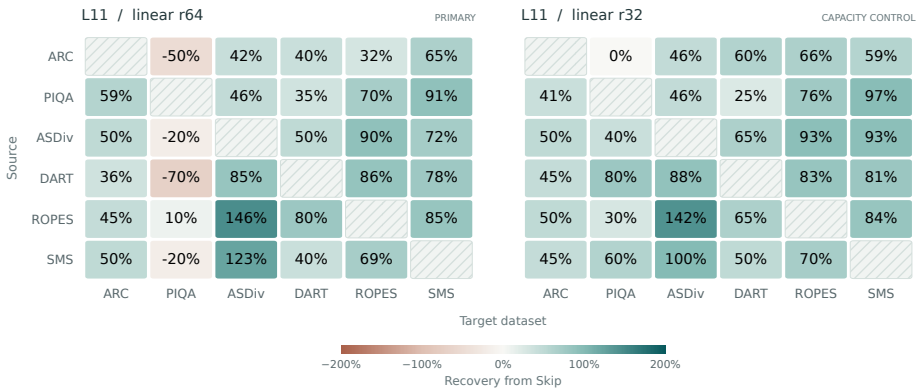


Figure 3(b): Llama transfer at L11. Results at the final layer are discussed in Section 4.3.2.

Reuse across task categories also succeeds in Qwen. At L17/r32, a surrogate trained on SMS recovers 106.7% of Skip damage on ASDiv, bringing accuracy above Dense (Figure 4). Such a classification source can thus support recovery on this mathematical task.

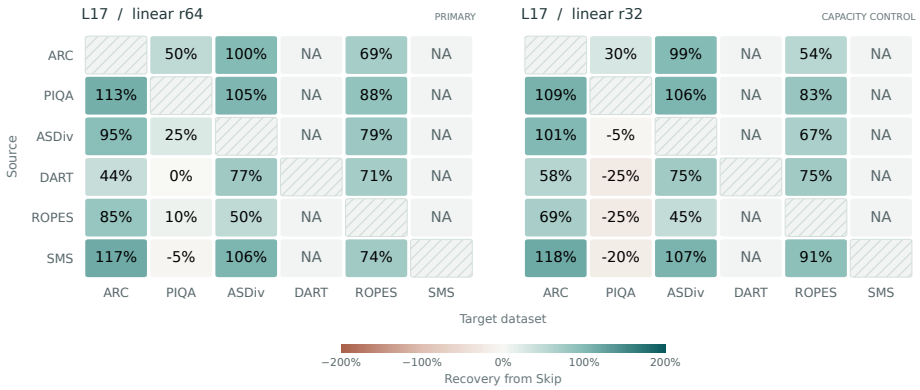


Figure 4: Qwen transfer at L17. Mean recovery (%) at r64 and r32; rows are sources and columns are targets. Color scale, NA convention and hatched diagonals match Figure 3. Each target has 100 examples (28 ROPES backgrounds); ROPES uses F1 and other tasks accuracy.

Across these examples, useful residual mappings can be learned from a different task category. The transfer grids reveal where reuse across tasks is effective.

4.3.2 Reuse at the final layer recovers QA performance

Surrogates trained on other tasks recover ROPES performance at the final layer. At Llama L31/r64, surrogates trained separately on ARC, PIQA, ASDiv, DART and SMS recover 70.9–81.6% of ROPES's Skip damage, with paired gain intervals above zero. At Qwen L35, surrogates trained on the same five datasets recover at least 91.4% at both tested ranks. Recovery across these datasets shows that replacement at the final layer can support ROPES without fitting on ROPES (Figure A3).

The same pattern extends to SQuAD, which is not used for Llama surrogate training. Averaged over all six training sources, replacement at the final layer recovers 95.7% of SQuAD's Skip damage and reaches F1 close to Dense. This demonstrates that surrogates at the final layer can preserve performance on question answering beyond the datasets used to fit them.

4.3.3 Transfer of replacements for layer pairs

Joint fitting can make replacements for layer pairs more transferable across tasks. The benefit seen in the comparison on the same dataset (Figure 2) extends to transfer between datasets, e.g., ROPES → SMS at Llama L2+L11: joint fitting improves over independent composition by 31.0 points at r64 and 33.0 points at r32. At both ranks, joint transfer recovers part of the Pair Skip loss, while independent composition remains below Pair Skip. Appendix D reports the complete transfer grids and paired intervals.

4.4 Ablation with random maps: training improves the surrogate

Fitting produces more effective replacements than calibrated random maps. As reported in Appendix B.2, trained r64 surrogates outperform the mean of random maps in all 18 evaluated settings. Outputs of random maps are scaled to the original layer's residual root mean square, providing a control with the original residual scale. Figure 5 compares task scores of trained surrogates with random maps on representative datasets. Figure B1 and Table B2 give full results on all datasets.

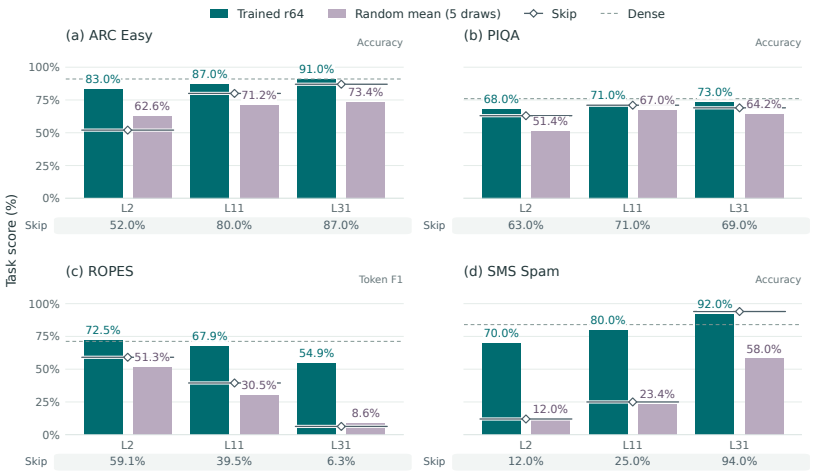


Figure 5: Trained surrogates and calibrated random maps at Llama L2/L11/L31. Bars show the trained r64 score and the mean of five random draws. Short horizontal lines with hollow diamonds mark Skip, with its scores listed below each panel; dashed lines mark Dense. Each task has 100 examples. Random maps match the original layer's residual root mean square. ROPES uses token F1 with 28 backgrounds; ARC, PIQA and SMS use accuracy.

Training yields substantial gains on sensitive settings: at SMS L2, the fitted surrogate reaches 70% accuracy while both Skip and the random mean score 12%; at ROPES L31, it improves over the random mean by 46.3 F1 points (Figure 5; Table B2). Fitting therefore turns a small replacement into a useful task update in settings where calibrated random maps recover little or no performance.

5 Discussion and Conclusion

Implications for compression. Recovery tests strengthen the necessity of replacements even for layers that are costly to remove (Section 4.2). This supports evaluating candidates through both Skip damage and recovery, choosing surrogate architecture by task performance, and testing joint fitting when replacing two layers together. Transfer across tasks offers reusable replacements when validated for the intended target and depth (Section 4.3). These choices expand compression beyond layers that tolerate removal.

Gains from skipping. Skip occasionally improves accuracy over Dense, as observed for SMS at Llama L31 (Table B2). PoLar also reports that searched execution programs using only layer skipping sometimes outperform the standard forward pass (Li et al., 2026). Explaining such gains is beyond the scope of this work, which focuses on recovering performance lost through layer removal. We will investigate when and why skipping improves performance in future work.

Limitations. This study examines the feasibility of replacing selected layers with surrogates that use a small fraction of their parameter budget. The reported compression ratios apply to the replaced layers; practical runtime and memory gains for the whole model have not been fully evaluated. Our experiments cover single layers and layer pairs, expecting testing replacements at more than two layers in our future work.

Conclusion. Our aim is to reduce layer parameters while preserving model performance. Skip, Replace and Transfer show that compact learned residual maps can recover sensitive layers, support reuse across tasks, and work together at more than one layer. These findings support compression guided by recovery, with surrogate architecture, layer interactions and transfer compatibility informing the choice of replacements.

AI USE STATEMENT

We used generative AI tools to assist with reviewing related literature and polishing the writing. We have reviewed all text produced with AI assistance and take responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

Section 2 defines the interventions and training objective; Appendix A.1 gives surrogate architectures, optimization settings and parameter counts; Appendix C gives a training example, the generation, parsing and scoring rules, the resampling procedures, and the protocols for layer selection and fitting. Code and data will be released upon publication.

REFERENCES

Tiago Almeida and José Hidalgo. SMS Spam Collection. UCI Machine Learning Repository, dataset, 2011. URL https://archive.ics.uci.edu/dataset/228/sms+spam+collection.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about Physical Commonsense in Natural Language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7432–7439, 2020. doi: 10.1609/aaai.v34i05.6239. URL https://ojs.aaai.org/index.php/AAAI/article/view/6239.

Xiaodong Chen, Yuxuan Hu, Jing Zhang, Yanling Wang, Cuiping Li, and Hong Chen. Streamlining Redundant Layers to Compress Large Language Models. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/4b00a351b41358965613c118e87dc28c-Abstract-Conference.html.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. arXiv preprint, 2018. URL https://arxiv.org/abs/1803.05457.

Elia Cunegatti, Marcus Vukojevic, Erik Nielsen, and Giovanni Iacca. From Layers to Submodules: Rethinking Granularity in Replacement-Based LLM Compression. arXiv preprint, 2026. URL https://arxiv.org/abs/2606.02559v2.

Aaron Grattafiori et al. The Llama 3 Herd of Models. arXiv preprint, 2024. URL https://arxiv.org/abs/2407.21783v3.

Andrey Gromov, Kushal Tirumala, Hassan Shapourian, Paolo Glorioso, and Daniel A. Roberts. The Unreasonable Ineffectiveness of the Deeper Layers. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2403.17887v2.

Cristian Hinostroza, Rodrigo Toro Icarte, Christ Devia, Andres Carvallo De Ferari, Eugenio Herrera-Berg, Denis Parra, and Jorge F. Silva. Rethinking Layer Relevance in Large Language Models Beyond Cosine Similarity. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2605.14075v1.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Tim Lawson and Laurence Aitchison. Learning to Skip the Middle Layers of Transformers. arXiv preprint, 2025. URL https://arxiv.org/abs/2506.21103v1.

Ziyue Li, Yang Li, and Tianyi Zhou. Skip a Layer or Loop It? Learning Program-of-Layers in LLMs. In *Forty-third International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=pl10b6EQAN.

Kevin Lin, Oyvind Tafjord, Peter Clark, and Matt Gardner. Reasoning Over Paragraph Effects in Situations. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pp. 58–62, 2019. doi: 10.18653/v1/D19-5808. URL https://aclanthology.org/D19-5808/.

Kainan Liu, Yong Zhang, Ning Cheng, Zhitao Li, Shaojun Wang, and Jing Xiao. GRASP: Replace Redundant Layers with Adaptive Singular Parameters for Efficient Model Compression. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 26333–26348, 2025. doi: 10.18653/v1/2025.emnlp-main.1338. URL https://aclanthology.org/2025.emnlp-main.1338/.

Xin Men, Mingyu Xu, Qingyu Zhang, Qianhao Yuan, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. ShortGPT: Layers in Large Language Models are More Redundant Than You Expect. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20192–20204, 2025. doi: 10.18653/v1/2025.findings-acl.1035. URL https://aclanthology.org/2025.findings-acl.1035/.

Shen-yun Miao, Chao-Chun Liang, and Keh-Yih Su. A Diverse Corpus for Evaluating and Developing English Math Word Problem Solvers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 975–984, 2020. doi: 10.18653/v1/2020.acl-main.92. URL https://aclanthology.org/2020.acl-main.92/.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, 2016. doi: 10.18653/v1/D16-1264. URL https://aclanthology.org/D16-1264/.

Anton Razzhigaev, Matvey Mikhalchuk, Elizaveta Goncharova, Nikolai Gerasimenko, Ivan Oseledets, Denis Dimitrov, and Andrey Kuznetsov. Your Transformer is Secretly Linear. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5376–5384, 2024. doi: 10.18653/v1/2024.acl-long.293. URL https://aclanthology.org/2024.acl-long.293/.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. FitNets: Hints for Thin Deep Nets. In *International Conference on Learning Representations (ICLR)*, 2015. URL https://arxiv.org/abs/1412.6550v4.

Dmitriy Shopkhoev, Ammar Ali, Magauiya Zhussip, Valentin Malykh, Stamatios Lefkimmiatis, Nikos Komodakis, and Sergey Zagoruyko. ReplaceMe: Network Simplification via Depth Pruning and Transformer Block Linearization. In *Advances in Neural Information Processing Systems 38 (NeurIPS)*, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/1c10d0c087c14689628124bbc8fa69f6-Paper-Conference.pdf.

Safal Shrestha, Anubhav Shrestha, Aadim Nepal, Minwu Kim, and Keith Ross. On the Limits of Layer Pruning for Generative Reasoning in Large Language Models. arXiv preprint, 2026. URL https://arxiv.org/abs/2602.01997v4.

Yuxuan Tong, Xiwen Zhang, Rui Wang, Ruidong Wu, and Junxian He. DART-Math: Difficulty-Aware Rejection Tuning for Mathematical Problem-Solving. In *Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/0ef1afa0daa888d695dcd5e9513bafa3-Abstract-Conference.html.

An Yang, Anfeng Li, Baosong Yang, et al. Qwen3 Technical Report. arXiv preprint, 2025. URL https://arxiv.org/abs/2505.09388.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level Convolutional Networks for Text Classification. In *Advances in Neural Information Processing Systems 28 (NIPS)*, 2015. URL https://papers.nips.cc/paper/2015/file/250cf8b51c773f3f8dc8b4be867a9a02-Paper.pdf.

A Additional Grids and Protocol Details

A.1 Distillation settings and parameter budgets

The Llama local capacity grid contains linear r32, r64 and r128, plus MLP64. The MLP has widths 4,096 $\rightarrow$ 64 $\rightarrow$ 4,096, SiLU activation and biases. It has 528,448 parameters, compared with 528,384 for linear r64. An affine map of rank r contains 2dr + d parameters. Relative to the Llama layer's 218,112,000 parameters, r32, r64 and r128 contain 266,240, 528,384 and 1,052,672 parameters, giving about 819$\times$, 413$\times$ and 207$\times$ parameter reductions. A Qwen layer contains 192,946,432 parameters. The MLP surrogate is $g_{\ell,m}^{\mathrm{MLP}}(h) = W_2\,\mathrm{SiLU}(W_1 h + b_1) + b_2$ with $W_1 \in \mathbb{R}^{m\times d}$ and $W_2 \in \mathbb{R}^{d\times m}$; like r, the width m sets the parameter budget, and its prediction is added to the layer input as in Equation (2). Because the teacher input h_ℓ^{dense} is shared, matching the residual in Equation (3) is equivalent to matching the complete layer output. Equation (3) weights token positions equally, so longer sequences contribute more terms. Residual reconstruction error and task scores measure different aspects of recovery and are evaluated separately.

The optimizer is AdamW with learning rate 0.001, zero weight decay, batches of 512 token vectors, and at most 50 epochs. Early stopping uses patience 5 and relative improvement 10^{-6}, selecting the checkpoint with the lowest validation loss. The reported residual error is mean squared vector error, rather than mean error per coordinate. Training uses scalar root mean square normalization computed only on source training data, with the scaling folded into exported weights.

A.2 Supplementary capacity and reconstruction analysis

Increasing linear rank reduces residual reconstruction MSE from r32 to r64 to r128 in every one of the 18 settings. Task damage does not follow a uniform ordering: 11 settings include at least one rank increase that raises damage despite better reconstruction, including 6 of the 12 sensitive settings (Figures A1 and A2). Section 4.2.1 summarizes this divergence.

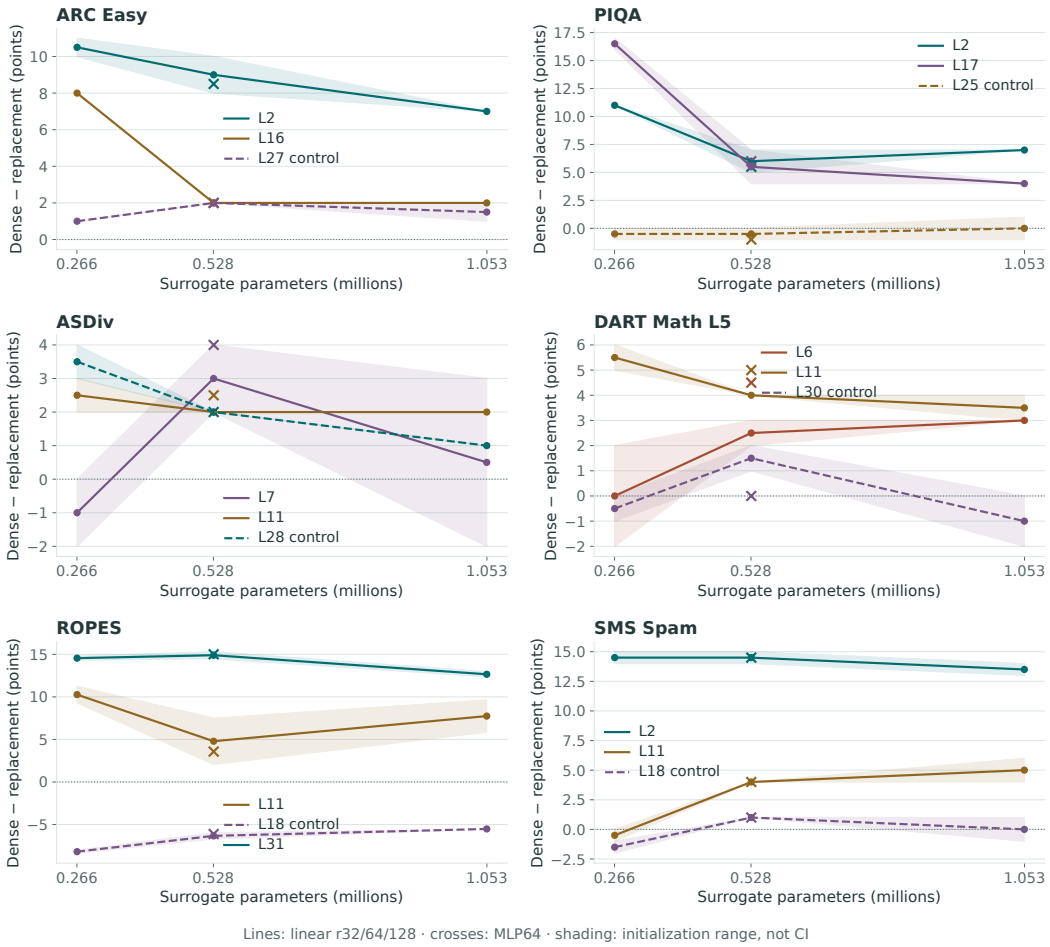


Figure A1: Capacity and architecture gains depend on the dataset and layer. Curves show all tested layers and the linear rank grid. Crosses denote MLP64, whose parameter count almost equals linear r64 (528,448 versus 528,384). MLP64 has lower damage in 5 of 18 settings, ties in 6 and higher damage in 7.

SMS L11 illustrates the disagreement directly. Residual MSE decreases from 4.877 to 3.820 to 3.417, while task damage increases from -0.5 to 4.0 to 5.0 points. Conversely, ARC L2 and PIQA L17 improve with rank. These contrasting trajectories show why task evaluation complements residual reconstruction loss.

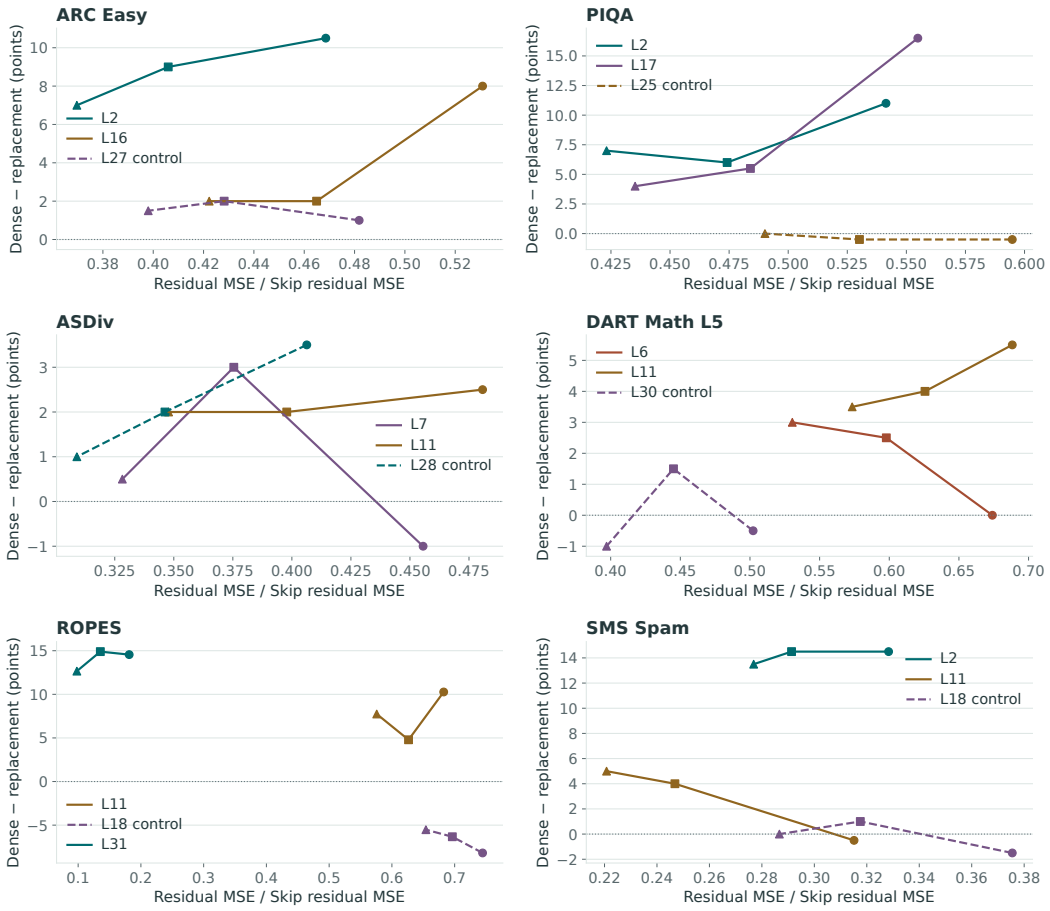


Figure A2: Lower reconstruction error need not improve task scores. Residual MSE is a mean squared vector error. Normalization uses the same layer's Skip reference with zero residual; it is not a measure of recovery in task performance. MSE falls strictly from r32 to r64 to r128 in all 18 settings, but 11 include a rank increase that raises damage, including 6 of 12 sensitive settings. For SMS L11, MSE falls 4.877→3.820→3.417 while damage rises −0.5→4.0→5.0 points.

Changing architecture also yields mixed outcomes. Compared with linear r64 at almost the same parameter count, MLP64 has lower mean damage in 5 settings, ties in 6 and higher damage in 7. DART L6 includes an MLP fit that only matches Skip. The architecture comparison captures additional variation under the tested training schedule and bottleneck width.

A.3 Complete Qwen removal and r64 recovery profiles

The removal scan (500 examples per task) and replacement evaluation (100 per task) use separate example sets. Figures A3 and A4 show the full removal scan and the r64 recovery profile; Figure A5 compares all tested architectures. Detailed scores for every combination of architecture, task and layer, together with reconstruction diagnostics, are in the supplementary material. The L6 comparison in Section 4.2.2 is one part of this grid.

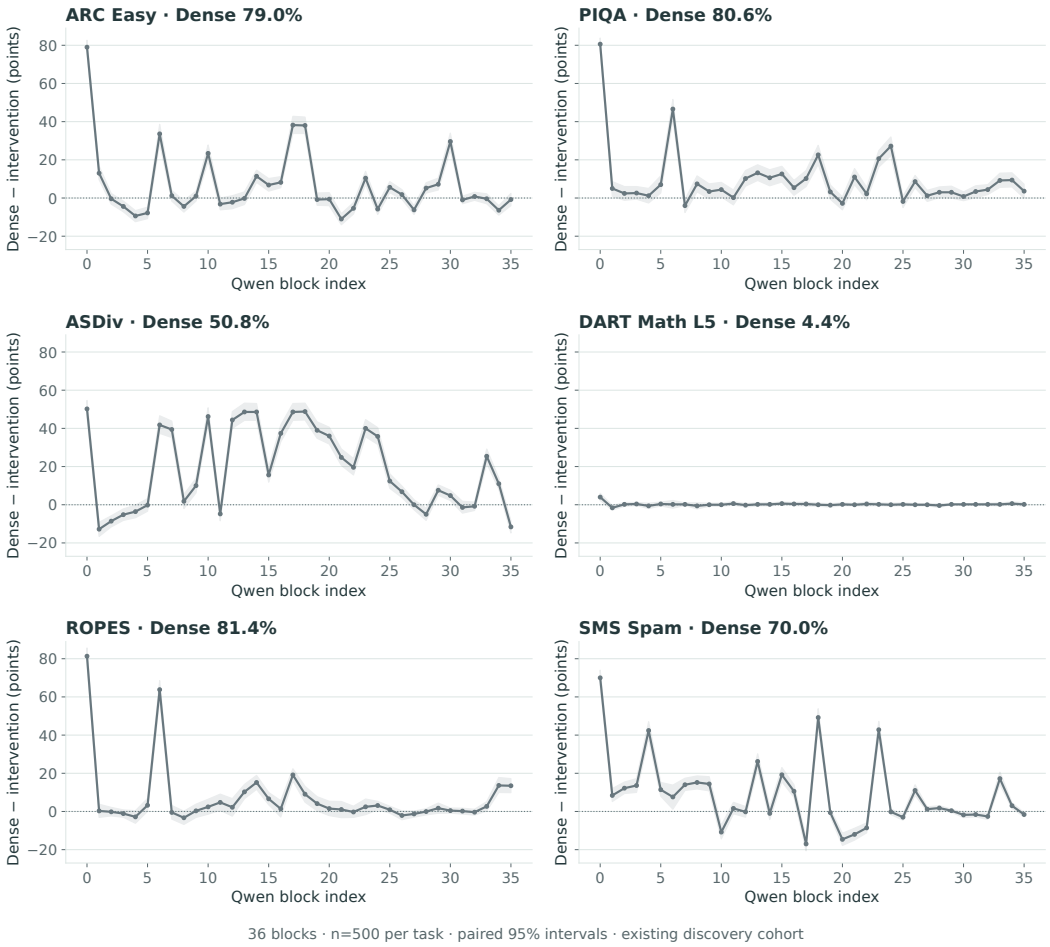


Figure A3: Qwen removal profile across all layers. Qwen3-8B with thinking disabled: 36 individual layer removals and Dense on 500 matched examples per task. L0 is most damaging on all six tasks; 61 of 216 removals raise the score. These examples are separate from those used in replacement and transfer evaluations, which each use 100 examples per task.

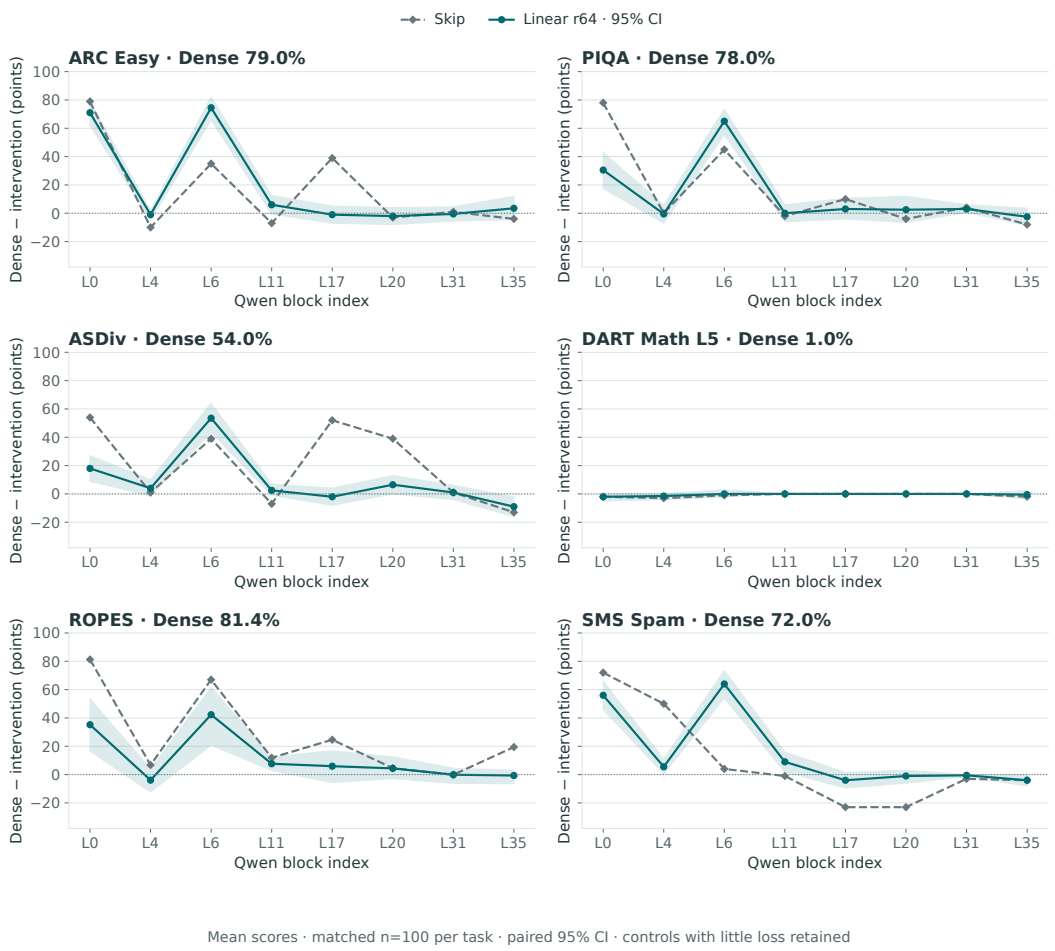


Figure A4: Qwen r64 recovery across all selected layers. Signed Dense−intervention score in points at eight layers. Gray denotes Skip and teal the r64 mean, with paired 95% intervals. Of 21 cells with Dense−Skip > 1 point, 17 improve; all four failures occur at L6.



Figure A5: Qwen recovery depends on surrogate architecture. Architecture and capacity comparison at the selected layers, using 100 test examples per task. At L6, MLP64 outperforms all three tested linear ranks on every dataset and restores ARC and PIQA to about Dense scores. Table 1 gives all six L6 comparisons.

The Qwen grid also includes ASDiv L17/MLP64 at 57.5% accuracy versus Dense 54% and Skip 2%, or 106.7% recovery. ROPES L4/r128 reaches 86.17% F1 versus Dense 81.40% and Skip 74.63%, or 170.3% recovery, with a gain of 11.53 points [5.46, 18.18]; its interval for the recovery ratio is NA because the denominator is unstable. DART has no eligible recovery denominator and a maximum mean score of 3.5% against Dense 1%. In this grid, 350 of 43,800 outputs are invalidated by thinking tags, 349 of them at L6.

A.4 Recovery and rank comparison at sensitive settings

The same local evaluation on six datasets also compares lower ranks. At ASDiv L7, linear r32 has 266,240 parameters, about 1/819 of the original layer, and reaches 17.0% mean accuracy against Dense 16.0% and Skip 2.0% on 100 examples, giving 107.1% mean recovery. Linear r64 reaches 13.0% accuracy, or 78.6% recovery (Table A1). Both results use the same test set. They describe a rank comparison within the evaluated grid.

Table A1: r64 recovery at every sensitive Llama setting.

Dataset	Layer	Skip damage	r64 damage [95% CI]	Recovery
ARC	L2	36.00	9.00 [3.50, 15.50]	75.0%
ARC	L16	16.00	2.00 [0.00, 5.00]	87.5%
ASDiv	L7	14.00	3.00 [−2.00, 8.50]	78.6%
ASDiv	L11	15.00	2.00 [−3.00, 8.00]	86.7%
DART	L6	5.00	2.50 [−3.50, 9.50]	50.0%
DART	L11	11.00	4.00 [−2.00, 10.50]	63.6%
PIQA	L2	12.00	6.00 [−3.50, 15.50]	50.0%
PIQA	L17	24.00	5.50 [−1.50, 12.50]	77.1%
ROPES	L11	31.34	4.79 [−3.13, 11.70]	84.7%
ROPES	L31	63.67	14.90 [4.47, 29.25]	76.6%
SMS	L2	72.00	14.50 [5.99, 23.51]	79.9%
SMS	L11	59.00	4.00 [0.00, 9.00]	93.2%

Damage is in points (ROPES: token F1; others: accuracy). Recovery is computed from mean scores. Every row uses r64 (528,384 parameters). Controls with little damage are in Figure 1.

B Complete Results for Layer Pairs and Random Controls

B.1 Complete joint replacement results for layer pairs

The following table reports all six tasks, both ranks and all four layer pairs. Each layer pair's independent and joint scores come from the same evaluation run; we never compare raw scores across runs. Detailed results and all intervals are in the supplementary material.

Table B1: Complete joint replacement comparison: every model, layer pair, task and rank.

Model / layer pair	Task	r	Dense (%)	Pair Skip (%)	Indep. (%)	Joint (%)	Joint − Indep. points [95% CI]	Recov. (%)
Llama L2 + L11	ARC	64	91.0	60.0	81.5	83.5	2.0 [-2.0, 6.5]	75.8
Llama L2 + L11	ARC	32	91.0	60.0	77.0	79.5	2.5 [-3.0, 8.0]	62.9
Llama L2 + L11	PIQA	64	76.0	55.0	63.5	66.5	3.0 [-3.5, 10.0]	54.8
Llama L2 + L11	PIQA	32	76.0	55.0	64.5	66.5	2.0 [-4.5, 9.0]	54.8
Llama L2 + L11	ASDiv	64	14.0	9.0	16.5	14.5	-2.0 [-5.5, 1.0]	110.0
Llama L2 + L11	ASDiv	32	14.0	9.0	13.5	11.5	-2.0 [-6.5, 2.5]	50.0
Llama L2 + L11	DART	64	13.0	4.0	11.5	13.5	2.0 [-3.0, 7.0]	105.6
Llama L2 + L11	DART	32	13.0	4.0	12.5	18.0	5.5 [-0.0, 12.0]	155.6
Llama L2 + L11	ROPES	64	71.2	33.6	56.6	61.1	4.5 [-3.9, 12.5]	73.1
Llama L2 + L11	ROPES	32	71.2	33.6	55.8	58.4	2.6 [-9.9, 13.1]	65.9

Continued on next page

Table B1 (continued)

Model / layer pair	Task	r	Dense (%)	Pair Skip (%)	Indep. (%)	Joint (%)	Joint − Indep. points [95% CI]	Recov. (%)
Llama L2 + L11	SMS	64	84.0	21.0	66.0	84.0	18.0 [11.0, 26.0]	100.0
Llama L2 + L11	SMS	32	84.0	21.0	65.0	83.5	18.5 [11.5, 26.0]	99.2
Llama L11 + L31	ARC	64	91.0	59.0	87.5	90.0	2.5 [-1.0, 6.0]	96.9
Llama L11 + L31	ARC	32	91.0	59.0	87.5	92.0	4.5 [1.5, 8.0]	103.1
Llama L11 + L31	PIQA	64	76.0	64.0	69.0	72.0	3.0 [0.0, 6.5]	66.7
Llama L11 + L31	PIQA	32	76.0	64.0	69.5	72.5	3.0 [-2.0, 8.5]	70.8
Llama L11 + L31	ASDiv	64	14.0	2.0	14.0	15.0	1.0 [-1.5, 4.0]	108.3
Llama L11 + L31	ASDiv	32	14.0	2.0	16.0	17.5	1.5 [-2.0, 5.0]	129.2
Llama L11 + L31	DART	64	13.0	4.0	7.5	12.5	5.0 [0.5, 9.5]	94.4
Llama L11 + L31	DART	32	13.0	4.0	12.5	11.5	-1.0 [-8.0, 5.5]	83.3
Llama L11 + L31	ROPES	64	71.2	4.8	50.0	52.4	2.4 [-1.8, 6.8]	71.6
Llama L11 + L31	ROPES	32	71.2	4.8	47.9	42.8	-5.0 [-11.6, 0.4]	57.2
Llama L11 + L31	SMS	64	84.0	88.0	90.0	93.0	3.0 [-1.5, 7.5]	NA
Llama L11 + L31	SMS	32	84.0	88.0	86.5	95.0	8.5 [3.5, 14.0]	NA
Qwen L4 + L17	ARC	64	79.0	63.0	83.5	83.5	0.0 [-1.5, 1.5]	128.1
Qwen L4 + L17	ARC	32	79.0	63.0	85.0	85.0	0.0 [0.0, 0.0]	137.5
Qwen L4 + L17	PIQA	64	78.0	71.0	79.0	63.5	-15.5 [-25.0, -6.0]	-107.1
Qwen L4 + L17	PIQA	32	78.0	71.0	74.0	70.5	-3.5 [-8.5, 1.5]	-7.1
Qwen L4 + L17	ASDiv	64	54.0	18.0	51.5	53.5	2.0 [-0.5, 5.0]	98.6
Qwen L4 + L17	ASDiv	32	54.0	18.0	48.5	52.0	3.5 [-1.0, 8.0]	94.4
Qwen L4 + L17	DART	64	1.0	1.0	1.0	1.0	0.0 [0.0, 0.0]	NA
Qwen L4 + L17	DART	32	1.0	1.0	1.0	1.0	0.0 [0.0, 0.0]	NA
Qwen L4 + L17	ROPES	64	81.4	57.0	73.8	73.2	-0.6 [-5.7, 4.0]	66.4
Qwen L4 + L17	ROPES	32	81.4	57.0	75.2	75.0	-0.2 [-5.0, 6.0]	73.7
Qwen L4 + L17	SMS	64	72.0	52.0	66.5	70.5	4.0 [1.5, 7.0]	92.5

Continued on next page

Table B1 (continued)

Model / layer pair	Task	r	Dense (%)	Pair Skip (%)	Indep. (%)	Joint (%)	Joint − Indep. points [95% CI]	Recov. (%)
Qwen L4 + L17	SMS	32	72.0	52.0	69.5	69.0	-0.5 [-2.5, 1.5]	85.0
Qwen L4 + L20	ARC	64	79.0	83.0	77.5	78.5	1.0 [0.0, 2.5]	NA
Qwen L4 + L20	ARC	32	79.0	83.0	79.5	80.0	0.5 [-2.0, 3.0]	NA
Qwen L4 + L20	PIQA	64	78.0	74.0	73.0	67.0	-6.0 [-17.0, 5.0]	-175.0
Qwen L4 + L20	PIQA	32	78.0	74.0	73.5	75.0	1.5 [-3.0, 6.5]	25.0
Qwen L4 + L20	ASDiv	64	54.0	20.0	42.0	48.5	6.5 [2.5, 11.0]	83.8
Qwen L4 + L20	ASDiv	32	54.0	20.0	39.5	49.5	10.0 [4.5, 16.0]	86.8
Qwen L4 + L20	DART	64	1.0	2.0	1.0	1.0	0.0 [0.0, 0.0]	NA
Qwen L4 + L20	DART	32	1.0	2.0	1.0	1.0	0.0 [0.0, 0.0]	NA
Qwen L4 + L20	ROPES	64	81.4	76.3	74.3	75.0	0.7 [-3.5, 4.8]	-27.1
Qwen L4 + L20	ROPES	32	81.4	76.3	80.6	80.4	-0.2 [-5.1, 4.9]	80.4
Qwen L4 + L20	SMS	64	72.0	38.0	69.5	70.5	1.0 [0.0, 2.5]	95.6
Qwen L4 + L20	SMS	32	72.0	38.0	70.5	70.5	0.0 [-2.0, 2.0]	95.6

Accuracy except ROPES (token F1). All tests have n=100; ROPES resamples 28 backgrounds. Paired intervals use 2,000 draws and are unadjusted. The supplementary material gives all score intervals and counts of unstable recovery estimates.

B.2 Every randomized residual comparison

The table lists all 18 comparisons. Confidence intervals condition on the fixed trained surrogate and random draws; ranges describe variation across the five random scores. The setup is in Section 3.4 and the results in Section 4.4; the construction of the random maps follows.

Construction. Random maps are surrogates whose parameters $U_{0,k}$, $V_{0,k}$, and $b_{0,k}$ are sampled at random, with the same shapes and rank. Let $\mathcal{V}_S$ denote the collected source validation token representations paired with the original layer residuals. For random draw k, define

$$q_k(h) = U_{0,k}V_{0,k}^{\top}h + b_{0,k},$$
$$g_{\mathrm{rand},k}(h) = \alpha_k q_k(h), \qquad \alpha_k = \sqrt{\frac{\sum_{(h,r)\in\mathcal{V}_S}\|r\|_2^2}{\sum_{(h,r)\in\mathcal{V}_S}\|q_k(h)\|_2^2}}. \tag{7}$$

The ratio of root mean squares is computed over the collected validation token vectors and their hidden coordinates. The scalar α_k is folded into $V_{0,k}$ and $b_{0,k}$, so inference uses the same affine residual form as Equation (2). The calibration matches the aggregate magnitude of the original layer's residual outputs, rather than parameter norms or directions at individual tokens. These random maps are not optimized to fit residuals or selected by task score. Once calibrated, their parameters remain fixed during evaluation, and the chosen layer is replaced throughout prefill and decode.

Let S_{fit} denote the trained surrogate's score and $S_{rand,k}$ the score for random draw k. We compare them through

$$\Delta = S_{\text{fit}} - \frac{1}{K}\sum_{k=1}^{K} S_{\text{rand},k}. \tag{8}$$

Here K is the number of random draws; positive Δ indicates an advantage for the trained surrogate over the mean random map.

Table B2: Llama ablation with random maps: all 18 comparisons.

Task	**Layer**	**Skip (%)**	**Trained (%)**	**Random mean (%)**	**Random range (%)**	**Trained advantage points [95% CI]**
ARC	L2	52.0	83.0	62.6	56.0–71.0	20.4 [13.6, 27.4]
ARC	L11	80.0	87.0	71.2	55.0–82.0	15.8 [9.4, 22.4]
ARC	L31	87.0	91.0	73.4	65.0–81.0	17.6 [12.8, 22.6]
PIQA	L2	63.0	68.0	51.4	45.0–62.0	16.6 [7.6, 25.4]
PIQA	L11	71.0	71.0	67.0	62.0–71.0	4.0 [-8.0, 15.8]
PIQA	L31	69.0	73.0	64.2	44.0–73.0	8.8 [2.6, 15.2]
ASDiv	L2	15.0	17.0	12.0	8.0–18.0	5.0 [-0.2, 10.8]
ASDiv	L11	1.0	15.0	1.8	1.0–2.0	13.2 [6.4, 20.6]
ASDiv	L31	17.0	16.0	10.0	6.0–15.0	6.0 [1.6, 11.0]
DART	L2	7.0	11.0	6.4	3.0–9.0	4.6 [-1.6, 11.0]
DART	L11	3.0	11.0	4.8	3.0–8.0	6.2 [0.0, 13.0]
DART	L31	12.0	14.0	6.0	3.0–8.0	8.0 [3.0, 13.6]
ROPES	L2	59.1	72.5	51.3	44.3–57.0	21.2 [11.9, 30.1]
ROPES	L11	39.5	67.9	30.5	12.2–44.2	37.4 [25.6, 50.0]
ROPES	L31	6.3	54.9	8.6	6.3–12.8	46.3 [30.7, 60.1]
SMS	L2	12.0	70.0	12.0	9.0–13.0	58.0 [48.4, 67.2]
SMS	L11	25.0	80.0	23.4	17.0–28.0	56.6 [47.0, 66.2]
SMS	L31	94.0	92.0	58.0	5.0–93.0	34.0 [28.8, 38.8]

Trained r64, five fixed random U/V/b draws and 100 test examples per task. The final column is the trained score minus the mean random score. Random residual outputs are calibrated to match the root mean square of the original layer's residual outputs. Ranges describe random draws; intervals resample paired test units.

Scale and random initialization also affect behavior. Relative to the unscaled map from draw 0, rescaling to the original residual root mean square lowers scores in 11 of 18 combinations, raises them in four and leaves three unchanged. SMS L31 random scores span 5–93%, compared with 92% for the trained surrogate and 94% for Skip. Trained residual root mean square is 0.614–0.977 times the original.

The BF16 calibration check gives root mean square ratios relative to the original residual of 0.9988–1.0007 across 90 random conditions and aggregate cosine similarities from -0.0189 to 0.0083. Figure 5 omits the ASDiv and DART panels for readability; Table B2 includes both.

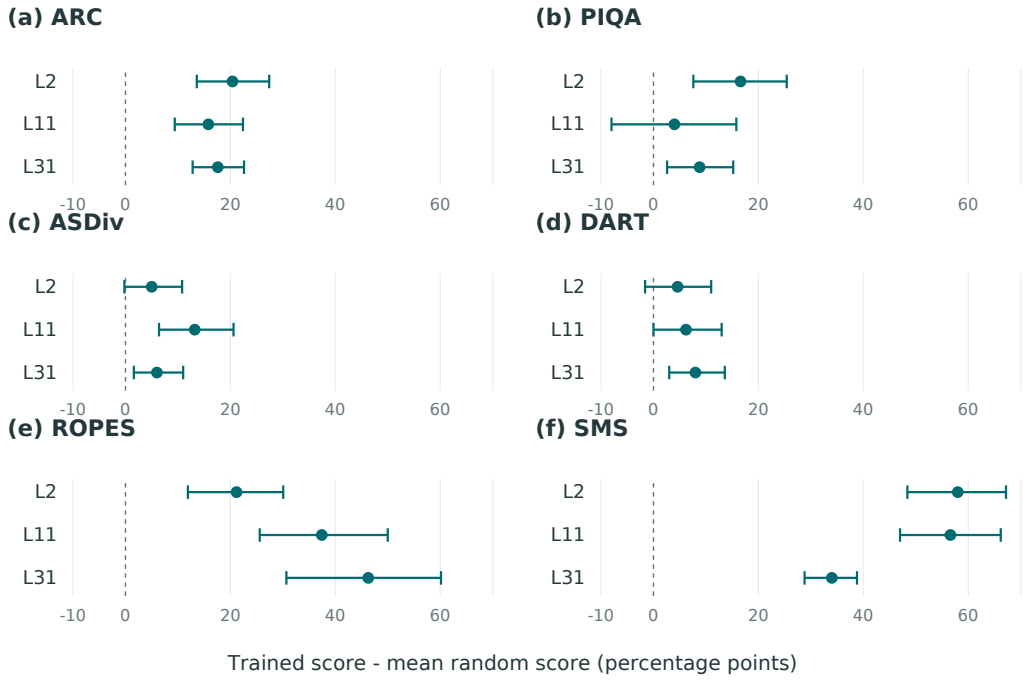


Figure B1: Paired uncertainty for all 18 Llama comparisons with random maps. Points show the trained r64 score minus the mean of five random maps calibrated to the original layer's residual root mean square; bars show paired 95% bootstrap intervals and dashed lines mark zero. All point estimates favor fitting, and 14 intervals exclude zero. Intervals condition on the fitted surrogate and five fixed random maps and resample 100 test examples per task (28 backgrounds for ROPES). These are the same evaluations as Figure 5 and Table B2, with ASDiv and DART also shown.

C Training Inputs and Evaluation Details

C.1 Training input example

The following prompt and reference are an actual ARC training record from the source split reused for replacement training. The text of the user message is shown verbatim; the reference is appended after the model's prefix for assistant generation when collecting teacher representations.

```
Answer the science question by choosing the correct option letter.

Question: Which of the following helps the leaves break down after
↪  they have fallen off the tree?
Choices:
A. meat-eaters
B. rocks
C. decomposers
D. cold temperatures

Answer:
```

Reference continuation:

```
C
```

For Llama, the user prompt is encoded with the native chat template and concatenated with the fixed reference tokens. Qwen uses its chat template with thinking disabled. Equation (3) supervises the layer's residual at every captured token position, including prompt and reference positions. The reference text therefore affects the teacher states even though the loss is residual MSE.

C.2 Generation, scoring and uncertainty

Generation is greedy, using fixed prompts and context settings. SQuAD and ROPES use maximum token F1 over reference answers; other tasks use accuracy. The label parser follows a fixed label priority, while the math parser extracts the first number with exact and fraction fallback and can misclassify answers containing multiple numbers. Accuracy and F1 are never pooled across tasks. Qwen's local and transfer evaluations score invalid outputs with thinking tags zero without retrying or removing samples. Each comparison uses matched Dense, Skip and intervention outputs from the same evaluation run. Each transfer grid uses a single software and numerical configuration, and we never combine scores from different evaluation runs, even when surrogate weights are reused.

Main Replace and Transfer estimates follow the averaging procedure in Section 3. Local recovery, transfer grids and comparisons of layer pairs use 2,000 draws, sampling examples or content groups and ROPES backgrounds while preserving the mean with equal weight per example. ROPES has 28 test backgrounds. The ablation with random maps uses the conditional procedure below. Intervals describe uncertainty from the test sample, with initialization ranges reported separately. Comparisons are exploratory and unadjusted for multiplicity, and intervals for the recovery ratio are NA when bootstrap denominators are unstable.

For the ablation with random maps, five random scores are averaged within each example before 2,000 paired draws resample the same test units for the trained surrogate and random maps. The interval is conditional on one trained surrogate and five fixed random maps; the range across draws describes random initialization variability separately. Transfer to AG News and SQuAD uses 100 AG News examples or 98 SQuAD content groups. Point recovery is NA when Dense$-$Skip is at most one point; a recovery interval is separately NA when bootstrap denominators are unstable. Neither status denotes a missing evaluation.

C.3 Selection, fitting and matched comparisons

Both checkpoints use hidden width 4,096 and BF16 evaluation; Qwen uses its chat template with thinking disabled. Llama has 32 layers and Qwen 36, so their final layers are L31 and L35; indices do not imply alignment between models.

For each Llama dataset in the suite of six tasks, the removal scan selects two layers with damage at least 0.01 and separated by at least two indices, ordered by descending damage with ties resolved deterministically by layer index. A control layer minimizes absolute removal damage among the remaining layers. This gives 12 sensitive and six control settings. Qwen replacement evaluates all six tasks at eight selected layers: L0/L4/L6/L11/L17/L20/L31/L35.

The local replacement protocol uses 300 training, 100 validation and 100 test sequences per dataset. All capacity and architecture comparisons in Appendix A use this split. Dense, Skip and replacement scores are matched within each evaluation.

Residual subtraction and AdamW training use FP32 with BF16 teacher states. Root mean square normalization is computed only from source training data and folded into exported weights. Evaluation casts the weights to the data type of the hidden states. Optimizer settings and parameter counts are in Appendix A.1.

Joint fitting starts from the matched independent weights. From the dense representation entering the first selected layer, the complete supervised sequence propagates through the first surrogate, frozen intervening layers and second surrogate. Both surrogates are optimized together to match the dense teacher's interval endpoint. The loss is normalized by the squared root mean square of the dense interval update on source training sequences. Gradients propagate through the frozen intervening computation, and source validation loss selects the checkpoint, including the initial weights as a candidate at epoch zero.

Joint optimization uses learning rate 0.001, zero weight decay, at most 50 epochs, patience 5, one sequence per microbatch, at least 512 tokens per update, gradient clipping at 1.0, FP32 master weights and BF16 forward computation. In the local replacement evaluation for layer pairs, each layer pair contributes 24 joint fits and 128 evaluated conditions, or 12,800 sample records; four layer pairs give 96 fits and 512 condition slots with reused references. The primary contrast is Joint$-$Independent on matched examples and seeds, while recovery uses simultaneous Skip as its denominator.

Each model's transfer evaluation for individual layers contains 72 surrogates and 456 conditions, including Dense and Skip references. Transfer of layer pairs reuses the fitted independent and joint weights without further training. Each layer pair has 288 surrogate conditions across sources, targets, ranks, training initializations and fitting modes plus 12 Dense and Pair Skip references: 300 conditions and 30,000 outputs per layer pair, or 1,200 conditions and 120,000 outputs across four layer pairs. All references, including local replacements for layer pairs, are reevaluated within this transfer run (Appendix D). Condition totals count repeated evaluation of the same examples, not additional independent observations.

DART Math L5 is the level 5 subset exported from the official DART-Math MATH pool after removing duplicate queries, using its query and reference answer fields. The example set and answer parser define the mathematical evaluation reported here.

D Complete Transfer Results for Layer Pairs

We extend frozen transfer to the same four layer pairs used for simultaneous Skip and replacement on the same dataset: Llama L2+L11 and L11+L31, and Qwen L4+L17 and L4+L20. Both modules are fitted on the row/source dataset. Independent modules retain their separately fitted weights; jointly fitted modules are transferred together. Neither weights nor normalization are fitted or recalibrated on a target. The existing trained weights are reused; this evaluation adds no training.

Each layer pair includes all six sources and six targets, ranks 64 and 32, and the training initializations specified in Section 3. Independent and joint conditions contribute $6 \times 6 \times 2 \times 2 \times 2 = 288$ evaluations per layer pair; six Dense and six Pair Skip references bring the total to 300 conditions and 30,000 generated outputs. Across four layer pairs this is 1,200 conditions and 120,000 outputs, comprising 240 comparisons across both ranks where source and target differ and 48 local comparisons of mean scores. These totals count repeated evaluation of the same examples.

Matched evaluation and uncertainty. All six targets retain 100 test examples. Dense, Pair Skip and local diagonal conditions are regenerated with the transfer conditions; each comparison uses this evaluation's matched references. The diagonal values are local references for this evaluation; Table B1 reports the separate local evaluation. We apply 2,000 paired bootstrap draws to the means within each example. ROPES uses 28 background groups and the other tasks use 100 fixed content groups each. ROPES scores are token F1; all other scores are accuracy. Scores are never pooled across targets. Generation, parsing and Qwen's rule for invalid thinking outputs follow Appendix C.2. Source token budgets vary, and joint fitting uses additional optimization. Intervals are exploratory and unadjusted for multiple comparisons.

Reading the grids. Each cell has three lines: Independent / Joint mean scores (%), followed by $\Delta_T = \text{Joint} - \text{Independent}$ in parentheses (points); the paired 95% interval for Δ_T; and the changes for initializations A / B, in a fixed order across all grids. Differences are computed before rounding. Bold Independent/Joint values mark local diagonal cells. Dense and Pair Skip references for each target appear below every grid. Positive Δ_T need not imply recovery to Dense or improvement over Pair Skip. The changes for each initialization describe initialization variability, not confidence intervals.

Of the 240 comparisons across both ranks where source and target differ, 50 have a paired change interval entirely above zero, 21 entirely below zero, and 169 include zero. These are descriptive counts; an interval containing zero does not establish equivalence.

D.1 LLAMA L2 + L11

Each cell: Independent / Joint scores followed by (Δ_T); paired 95% CI; changes for initializations A / B. Bold Independent/Joint values identify local diagonals.

Table D1: Llama L2 + L11: complete r64 transfer grid.

Source	ARC	PIQA	ASDiv	DART	ROPES	SMS
	81.0/**84.0**(3.0)	62.0/68.0(6.0)	19.5/13.5(−6.0)	11.0/14.5(3.5)	55.1/52.8(−2.4)	12.0/15.5(3.5)
	[−1.0, 7.5]	[−3.5, 15.5]	[−11.5, −1.5]	[−1.5, 8.5]	[−11.5, 4.4]	[0.5, 7.5]
ARC	+3.0/ + 3.0	+7.0/ + 5.0	−6.0/ − 6.0	+6.0/ + 1.0	−3.0/ − 1.7	+4.0/ + 3.0
	74.0/79.0(5.0)	**63.5**/**66.5**(3.0)	21.5/21.0(−0.5)	14.5/14.0(−0.5)	53.2/66.7(13.5)	11.0/24.5(13.5)
	[0.0, 10.5]	[−3.5, 10.0]	[−6.0, 5.0]	[−3.5, 3.0]	[5.5, 20.2]	[7.5, 20.5]
PIQA	+7.0/ + 3.0	+6.0/0.0	+1.0/ − 2.0	−1.0/0.0	+15.1/ + 11.9	+15.0/ + 12.0
	68.5/77.0(8.5)	47.5/59.5(12.0)	**16.5**/**14.5**(−2.0)	10.5/12.0(1.5)	52.8/57.8(5.0)	12.0/15.5(3.5)
	[2.5, 15.0]	[0.0, 23.5]	[−5.5, 1.0]	[−3.5, 6.5]	[−5.6, 14.4]	[1.0, 7.0]
ASDiv	+10.0/ + 7.0	+10.0/ + 14.0	−3.0/ − 1.0	+2.0/ + 1.0	+7.8/ + 2.2	+5.0/ + 2.0
	59.0/68.5(9.5)	46.5/63.0(16.5)	16.0/15.5(−0.5)	**11.5**/**13.5**(2.0)	56.3/62.9(6.6)	47.5/20.5(−27.0)
	[3.0, 16.0]	[7.0, 27.0]	[−5.0, 4.0]	[−3.0, 7.0]	[−2.0, 16.2]	[−36.5, −17.0]
DART	+10.0/ + 9.0	+15.0/ + 18.0	+1.0/ − 2.0	+1.0/ + 3.0	+6.7/ + 6.4	−24.0/ − 30.0
	76.5/80.5(4.0)	59.0/70.5(11.5)	41.0/39.0(−2.0)	14.5/11.5(−3.0)	**56.6**/**61.1**(4.5)	17.0/48.0(31.0)
	[−1.0, 9.5]	[1.5, 21.0]	[−7.5, 3.0]	[−7.5, 1.0]	[−3.9, 12.5]	[22.5, 39.5]
ROPES	+4.0/ + 4.0	+11.0/ + 12.0	−4.0/0.0	−3.0/ − 3.0	+7.2/ + 1.8	+28.0/ + 34.0
	65.0/74.0(9.0)	65.0/69.0(4.0)	39.5/32.5(−7.0)	11.0/11.0(0.0)	61.4/65.0(3.7)	**66.0**/**84.0**(18.0)
	[3.5, 14.5]	[−4.5, 12.5]	[−13.0, −1.5]	[−5.0, 4.5]	[−2.6, 8.9]	[11.0, 26.0]
SMS	+8.0/ + 10.0	+7.0/ + 1.0	−8.0/ − 6.0	0.0/0.0	+2.7/ + 4.6	+17.0/ + 19.0
Dense	91.0	76.0	14.0	13.0	71.2	84.0
Pair Skip	61.0	55.0	9.0	4.0	33.6	21.0

Table D2: Llama L2 + L11: complete r32 transfer grid.

Source	ARC	PIQA	ASDiv	DART	ROPES	SMS
	77.5/**79.5**(2.0)	58.0/69.5(11.5)	14.5/15.5(1.0)	9.5/7.5(−2.0)	59.9/58.3(−1.7)	13.5/14.5(1.0)
	[−3.0, 7.0]	[1.5, 21.0]	[−2.5, 4.5]	[−5.5, 0.5]	[−11.1, 7.8]	[−1.5, 4.0]
ARC	0.0/ + 4.0	+12.0/ + 11.0	0.0/ + 2.0	−2.0/ − 2.0	−2.3/ − 1.1	0.0/ + 2.0
	71.5/78.5(7.0)	**64.5**/**66.5**(2.0)	16.0/23.5(7.5)	15.0/13.0(−2.0)	53.2/64.5(11.3)	11.0/17.5(6.5)
	[0.5, 14.0]	[−4.5, 9.0]	[3.5, 12.5]	[−6.5, 2.0]	[0.5, 21.0]	[2.5, 11.5]
PIQA	+7.0/ + 7.0	+2.0/ + 2.0	+9.0/ + 6.0	−3.0/ − 1.0	+12.5/ + 10.0	+6.0/ + 7.0
	66.5/75.5(9.0)	56.5/60.5(4.0)	**13.5**/**11.5**(−2.0)	12.5/10.5(−2.0)	56.6/60.0(3.4)	12.0/18.0(6.0)
	[3.0, 15.5]	[−5.5, 14.0]	[−6.5, 2.5]	[−5.0, 1.0]	[−5.5, 11.8]	[2.0, 11.0]
ASDiv	+9.0/ + 9.0	+4.0/ + 4.0	−2.0/ − 2.0	−2.0/ − 2.0	+6.1/ + 0.7	+5.0/ + 7.0
	57.5/70.0(12.5)	50.0/61.5(11.5)	25.5/23.5(−2.0)	**12.5**/**18.0**(5.5)	58.6/61.1(2.5)	35.0/25.5(−9.5)
	[5.0, 20.5]	[1.0, 21.5]	[−7.0, 2.0]	[0.0, 12.0]	[−6.2, 11.3]	[−17.5, −1.5]
DART	+14.0/ + 11.0	+15.0/ + 8.0	−2.0/ − 2.0	+5.0/ + 6.0	+4.1/ + 1.0	−14.0/ − 5.0
	73.0/77.5(4.5)	62.5/66.5(4.0)	40.5/45.5(5.0)	11.5/12.0(0.5)	**55.8**/**58.4**(2.6)	16.0/49.0(33.0)
	[−1.5, 11.0]	[−5.0, 13.5]	[−0.5, 11.0]	[−4.5, 6.0]	[−9.9, 13.1]	[24.0, 42.0]
ROPES	+5.0/ + 4.0	+6.0/ + 2.0	+4.0/ + 6.0	−1.0/ + 2.0	+3.0/ + 2.2	+31.0/ + 35.0
	65.5/75.5(10.0)	69.0/72.5(3.5)	33.5/30.0(−3.5)	10.5/10.5(0.0)	61.9/67.2(5.3)	**65.0**/**83.5**(18.5)
	[1.5, 18.5]	[−5.0, 12.0]	[−7.5, 0.0]	[−4.0, 4.0]	[−4.6, 15.5]	[11.5, 26.0]
SMS	+12.0/ + 8.0	+2.0/ + 5.0	−3.0/ − 4.0	0.0/0.0	+5.2/ + 5.4	+19.0/ + 18.0
Dense	91.0	76.0	14.0	13.0	71.2	84.0
Pair Skip	61.0	55.0	9.0	4.0	33.6	21.0

D.2 LLAMA L11 + L31

Each cell: Independent / Joint scores followed by (Δ_T); paired 95% CI; changes for initializations A / B. Bold Independent/Joint values identify local diagonals.

Table D3: Llama L11 + L31: complete r64 transfer grid.

Source	ARC	PIQA	ASDiv	DART	ROPES	SMS
	87.5/90.0(2.5)	59.5/69.5(10.0)	1.0/1.0(0.0)	5.0/4.5(−0.5)	35.6/32.7(−2.9)	93.0/91.5(−1.5)
	[−1.0, 6.0]	[2.0, 18.5]	[0.0, 0.0]	[−2.5, 1.5]	[−7.6, 1.2]	[−4.0, 0.0]
ARC	+3.0/ + 2.0	+12.0/ + 8.0	0.0/0.0	0.0/ − 1.0	−2.6/ − 3.2	−1.0/ − 2.0
	84.5/87.0(2.5)	**69.0/72.0**(3.0)	2.0/7.0(5.0)	4.5/5.5(1.0)	40.3/40.2(−0.1)	93.5/92.5(−1.0)
	[−1.5, 6.5]	[0.0, 6.5]	[2.0, 8.5]	[−1.0, 3.0]	[−5.4, 3.9]	[−5.0, 3.0]
PIQA	+3.0/ + 2.0	+3.0/ + 3.0	+7.0/ + 3.0	0.0/ + 2.0	−0.2/ − 0.1	0.0/ − 2.0
	80.0/79.5(−0.5)	58.5/60.0(1.5)	**14.0/15.0**(1.0)	12.0/15.0(3.0)	41.2/41.6(0.4)	91.0/88.0(−3.0)
	[−4.5, 3.5]	[−2.5, 6.0]	[−1.5, 4.0]	[−1.0, 7.0]	[−3.6, 4.6]	[−6.5, 0.0]
ASDiv	0.0/ − 1.0	+3.0/0.0	+2.0/0.0	+2.0/ + 4.0	+0.7/ + 0.1	−4.0/ − 2.0
	76.0/79.5(3.5)	65.0/57.5(−7.5)	9.5/26.5(17.0)	**7.5/12.5**(5.0)	40.3/49.8(9.6)	91.5/94.5(3.0)
	[−0.5, 8.0]	[−15.5, 0.5]	[9.5, 25.5]	[0.5, 9.5]	[5.0, 14.9]	[−1.0, 7.5]
DART	+1.0/ + 6.0	−5.0/ − 10.0	+17.0/ + 17.0	+7.0/ + 3.0	+12.4/ + 6.8	+4.0/ + 2.0
	86.0/89.5(3.5)	70.5/62.5(−8.0)	19.5/20.0(0.5)	8.5/5.5(−3.0)	**50.0/52.4**(2.4)	94.0/94.0(0.0)
	[0.5, 7.5]	[−15.0, −1.0]	[−6.5, 8.0]	[−6.5, 0.0]	[−1.8, 6.8]	[−5.0, 5.0]
ROPES	+3.0/ + 4.0	−7.0/ − 9.0	0.0/ + 1.0	−4.0/ − 2.0	+2.5/ + 2.2	0.0/0.0
	80.5/80.5(0.0)	60.0/60.5(0.5)	17.5/9.5(−8.0)	5.0/4.5(−0.5)	42.6/35.3(−7.3)	**90.0/93.0**(3.0)
	[−5.5, 5.5]	[−4.5, 5.5]	[−13.0, −3.5]	[−3.5, 2.0]	[−13.8, −1.5]	[−1.5, 7.5]
SMS	0.0/0.0	+1.0/0.0	−11.0/ − 5.0	−2.0/ + 1.0	−8.6/ − 6.1	+3.0/ + 3.0
Dense	91.0	76.0	14.0	13.0	71.2	84.0
Pair Skip	59.0	64.0	2.0	4.0	4.8	88.0

Table D4: Llama L11 + L31: complete r32 transfer grid.

Source	ARC	PIQA	ASDiv	DART	ROPES	SMS
	87.5/92.0(4.5)	66.5/70.0(3.5)	3.0/2.0(−1.0)	5.5/9.0(3.5)	41.0/38.0(−3.0)	92.5/91.0(−1.5)
	[1.5, 8.0]	[−5.5, 12.5]	[−4.0, 1.0]	[−0.5, 8.0]	[−5.4, 0.0]	[−5.5, 2.0]
ARC	+5.0/ + 4.0	+5.0/ + 2.0	0.0/ − 2.0	+2.0/ + 5.0	−1.8/ − 4.2	−1.0/ − 2.0
	83.5/84.0(0.5)	**69.5/72.5**(3.0)	3.0/5.0(2.0)	5.5/7.5(2.0)	38.3/40.2(1.9)	94.0/89.0(−5.0)
	[−4.5, 5.5]	[−2.0, 8.5]	[−1.5, 6.0]	[−1.5, 5.5]	[−1.7, 4.7]	[−10.0, −0.5]
PIQA	+2.0/ − 1.0	+4.0/ + 2.0	+4.0/0.0	0.0/ + 4.0	+1.4/ + 2.4	−4.0/ − 6.0
	79.0/83.0(4.0)	59.5/57.5(−2.0)	**16.0/17.5**(1.5)	13.0/15.0(2.0)	42.8/44.3(1.5)	93.0/91.5(−1.5)
	[0.0, 9.0]	[−7.5, 3.0]	[−2.0, 5.0]	[−2.0, 6.0]	[−3.4, 7.3]	[−4.0, 0.0]
ASDiv	+4.0/ + 4.0	−2.0/ − 2.0	+1.0/ + 2.0	+3.0/ + 1.0	+0.7/ + 2.3	−2.0/ − 1.0
	76.0/78.5(2.5)	64.5/60.5(−4.0)	11.5/27.0(15.5)	**12.5/11.5**(−1.0)	46.4/53.4(7.0)	93.0/94.0(1.0)
	[−3.0, 8.0]	[−11.0, 3.0]	[9.0, 22.5]	[−8.0, 5.5]	[1.8, 12.2]	[−4.0, 6.0]
DART	+3.0/ + 2.0	−5.0/ − 3.0	+15.0/ + 16.0	0.0/ − 2.0	+8.9/ + 5.1	+2.0/0.0
	86.5/87.0(0.5)	68.5/62.5(−6.0)	19.0/6.0(−13.0)	8.0/5.5(−2.5)	**47.9/42.8**(−5.0)	94.0/92.0(−2.0)
	[−3.5, 4.5]	[−12.0, −0.5]	[−20.0, −7.0]	[−6.0, 0.5]	[−11.6, 0.4]	[−5.0, 1.0]
ROPES	−1.0/ + 2.0	−4.0/ − 8.0	−11.0/ − 15.0	−2.0/ − 3.0	−3.6/ − 6.5	−3.0/ − 1.0
	80.0/79.0(−1.0)	64.5/62.0(−2.5)	11.5/8.5(−3.0)	7.0/4.5(−2.5)	43.0/44.5(1.5)	**86.5/95.0**(8.5)
	[−6.0, 4.5]	[−7.5, 2.0]	[−9.0, 3.0]	[−6.5, 1.0]	[−1.2, 4.5]	[3.5, 14.0]
SMS	−1.0/ − 1.0	−3.0/ − 2.0	−1.0/ − 5.0	−1.0/ − 4.0	−2.6/ + 5.5	+7.0/ + 10.0
Dense	91.0	76.0	14.0	13.0	71.2	84.0
Pair Skip	59.0	64.0	2.0	4.0	4.8	88.0

D.3 QWEN L4 + L17

Each cell: Independent / Joint scores followed by (Δ_T); paired 95% CI; changes for initializations A / B. Bold Independent/Joint values identify local diagonals.

Table D5: Qwen L4 + L17: complete r64 transfer grid.

Source	ARC	PIQA	ASDiv	DART	ROPES	SMS
	83.5/**83.5**(0.0)	73.0/72.5(−0.5)	57.5/58.5(1.0)	1.0/1.0(0.0)	72.6/72.5(−0.1)	13.0/13.5(0.5)
	[−1.5, 1.5]	[−2.0, 1.0]	[0.0, 2.5]	[0.0, 0.0]	[−2.2, 1.7]	[0.0, 1.5]
ARC	+1.0/ − 1.0	−1.0/0.0	+2.0/0.0	0.0/0.0	+0.1/ − 0.3	+1.0/0.0
	74.0/63.5(−10.5)	**79.0**/**63.5**(−15.5)	47.5/59.0(11.5)	3.0/6.0(3.0)	77.1/73.7(−3.4)	25.0/25.0(0.0)
	[−17.0, −4.5]	[−25.0, −6.0]	[3.5, 19.5]	[0.0, 6.5]	[−11.0, 4.4]	[−4.0, 4.0]
PIQA	−8.0/ − 13.0	−18.0/ − 13.0	+11.0/ + 12.0	+3.0/ + 3.0	−6.8/0.0	−6.0/ + 6.0
	73.5/79.5(6.0)	64.5/68.0(3.5)	**51.5**/**53.5**(2.0)	1.0/1.0(0.0)	77.3/77.2(0.0)	32.5/47.5(15.0)
	[2.5, 10.5]	[−0.5, 7.5]	[−0.5, 5.0]	[0.0, 0.0]	[−4.0, 4.6]	[10.0, 20.5]
ASDiv	+6.0/ + 6.0	+3.0/ + 4.0	+1.0/ + 3.0	0.0/0.0	−1.1/ + 1.0	+12.0/ + 18.0
	61.5/61.0(−0.5)	70.5/70.0(−0.5)	23.0/23.5(0.5)	**1.0**/**1.0**(0.0)	71.9/73.2(1.2)	13.0/13.0(0.0)
	[−1.5, 0.0]	[−2.5, 1.0]	[0.0, 1.5]	[0.0, 0.0]	[0.0, 3.3]	[0.0, 0.0]
DART	−1.0/0.0	−1.0/0.0	+1.0/0.0	0.0/0.0	+0.5/ + 2.0	0.0/0.0
	73.0/78.5(5.5)	67.5/65.5(−2.0)	39.5/22.0(−17.5)	1.0/1.0(0.0)	**73.8**/**73.2**(−0.6)	29.0/42.5(13.5)
	[1.5, 10.0]	[−7.0, 2.5]	[−22.5, −13.0]	[0.0, 0.0]	[−5.7, 4.0]	[9.5, 18.0]
ROPES	+11.0/0.0	−4.0/0.0	−35.0/0.0	0.0/0.0	−1.2/0.0	+27.0/0.0
	80.0/82.0(2.0)	62.5/62.5(0.0)	54.5/53.0(−1.5)	2.0/2.5(0.5)	75.5/77.1(1.6)	**66.5**/**70.5**(4.0)
	[0.0, 4.5]	[−3.0, 3.0]	[−4.0, 1.0]	[0.0, 1.5]	[−0.6, 4.9]	[1.5, 7.0]
SMS	+3.0/ + 1.0	−1.0/ + 1.0	−2.0/ − 1.0	+1.0/0.0	+2.5/ + 0.7	+7.0/ + 1.0
Dense	79.0	78.0	54.0	1.0	81.4	72.0
Pair Skip	63.0	71.0	18.0	1.0	57.0	52.0

Table D6: Qwen L4 + L17: complete r32 transfer grid.

Source	ARC	PIQA	ASDiv	DART	ROPES	SMS
	85.0/**85.0**(0.0)	74.0/74.0(0.0)	58.0/58.0(0.0)	2.0/2.0(0.0)	73.4/73.4(0.0)	13.5/13.5(0.0)
	[0.0, 0.0]	[0.0, 0.0]	[0.0, 0.0]	[0.0, 0.0]	[0.0, 0.0]	[0.0, 0.0]
ARC	0.0/0.0	0.0/0.0	0.0/0.0	0.0/0.0	0.0/0.0	0.0/0.0
	78.0/79.0(1.0)	**74.0**/**70.5**(−3.5)	53.0/55.0(2.0)	3.5/3.5(0.0)	79.2/79.0(−0.2)	21.0/25.0(4.0)
	[−1.5, 3.5]	[−8.5, 1.5]	[−3.0, 6.5]	[−1.5, 1.5]	[−1.6, 1.4]	[1.5, 6.5]
PIQA	+4.0/ − 2.0	−6.0/ − 1.0	+3.0/ + 1.0	0.0/0.0	−1.5/ + 1.0	+8.0/0.0
	80.0/83.0(3.0)	68.5/67.0(−1.5)	**48.5**/**52.0**(3.5)	1.0/1.0(0.0)	76.6/77.2(0.5)	34.5/40.0(5.5)
	[0.5, 6.5]	[−8.5, 5.5]	[−1.0, 8.0]	[0.0, 0.0]	[−2.9, 4.3]	[1.5, 10.0]
ASDiv	+4.0/ + 2.0	−3.0/0.0	+3.0/ + 4.0	0.0/0.0	−0.6/ + 1.7	+4.0/ + 7.0
	71.5/67.0(−4.5)	71.0/70.5(−0.5)	16.5/12.5(−4.0)	**1.0**/**1.0**(0.0)	73.9/76.5(2.5)	13.0/13.5(0.5)
	[−8.5, −1.0]	[−5.5, 4.5]	[−7.0, −1.0]	[0.0, 0.0]	[−1.3, 7.7]	[−1.0, 2.5]
DART	0.0/ − 9.0	0.0/ − 1.0	0.0/ − 8.0	0.0/0.0	0.0/ + 5.1	0.0/ + 1.0
	79.0/86.5(7.5)	68.0/65.0(−3.0)	28.5/29.5(1.0)	1.0/2.0(1.0)	**75.2**/**75.0**(−0.2)	36.0/76.5(40.5)
	[1.5, 14.5]	[−11.0, 5.0]	[−5.0, 7.5]	[0.0, 2.5]	[−5.0, 6.0]	[32.0, 49.0]
ROPES	+7.0/ + 8.0	−6.0/0.0	+1.0/ + 1.0	+2.0/0.0	−1.3/ + 0.9	+31.0/ + 50.0
	84.0/84.5(0.5)	63.5/61.5(−2.0)	54.5/56.0(1.5)	2.5/3.5(1.0)	81.3/81.4(0.1)	**69.5**/**69.0**(−0.5)
	[−1.0, 2.5]	[−5.5, 1.5]	[−2.0, 4.5]	[0.0, 2.5]	[−2.1, 2.1]	[−2.5, 1.5]
SMS	+1.0/0.0	−3.0/ − 1.0	+4.0/ − 1.0	+2.0/0.0	−0.8/ + 1.0	−1.0/0.0
Dense	79.0	78.0	54.0	1.0	81.4	72.0
Pair Skip	63.0	71.0	18.0	1.0	57.0	52.0

D.4 QWEN L4 + L20

Each cell: Independent / Joint scores followed by (Δ_T); paired 95% CI; changes for initializations A / B. Bold Independent/Joint values identify local diagonals.

Table D7: Qwen L4 + L20: complete r64 transfer grid.

Source	ARC	PIQA	ASDiv	DART	ROPES	SMS
	77.5/**78.5**(1.0)	75.0/77.5(2.5)	42.0/42.0(0.0)	1.0/1.0(0.0)	79.7/80.5(0.8)	32.0/39.0(7.0)
	[0.0, 2.5]	[0.0, 5.5]	[−1.5, 1.5]	[0.0, 0.0]	[−1.3, 3.6]	[4.0, 10.5]
ARC	+2.0/0.0	+4.0/ + 1.0	+1.0/ − 1.0	0.0/0.0	+1.0/ + 0.7	+14.0/0.0
	84.0/84.5(0.5)	**73.0**/**67.0**(−6.0)	50.5/43.5(−7.0)	2.5/2.5(0.0)	80.5/76.2(−4.2)	81.0/60.0(−21.0)
	[−4.0, 5.0]	[−17.0, 5.0]	[−13.5, −1.0]	[0.0, 0.0]	[−10.8, 1.3]	[−27.5, −14.5]
PIQA	+2.0/ − 1.0	−10.0/ − 2.0	−5.0/ − 9.0	0.0/0.0	−5.0/ − 3.5	−24.0/ − 18.0
	51.5/65.5(14.0)	78.5/80.5(2.0)	**42.0**/**48.5**(6.5)	2.5/3.0(0.5)	78.9/80.5(1.6)	56.5/62.5(6.0)
	[8.5, 20.5]	[−1.0, 5.5]	[2.5, 11.0]	[0.0, 1.5]	[−2.5, 6.3]	[2.5, 9.5]
ASDiv	+19.0/ + 9.0	+3.0/ + 1.0	+8.0/ + 5.0	+1.0/0.0	−0.8/ + 4.0	+7.0/ + 5.0
	50.0/50.0(0.0)	76.5/77.0(0.5)	39.0/38.5(−0.5)	**1.0**/**1.0**(0.0)	77.3/77.6(0.2)	24.5/25.0(0.5)
	[−1.5, 1.5]	[0.0, 1.5]	[−2.5, 1.0]	[0.0, 0.0]	[−1.4, 2.6]	[−1.0, 2.0]
DART	+1.0/ − 1.0	0.0/ + 1.0	0.0/ − 1.0	0.0/0.0	+0.5/0.0	0.0/ + 1.0
	76.0/77.0(1.0)	71.0/75.5(4.5)	15.0/12.5(−2.5)	2.5/2.0(−0.5)	**74.3**/**75.0**(0.7)	78.5/88.0(9.5)
	[−1.0, 3.0]	[0.5, 8.5]	[−6.5, 1.5]	[−2.0, 1.0]	[−3.5, 4.8]	[5.5, 13.5]
ROPES	+2.0/0.0	+9.0/0.0	−5.0/0.0	−1.0/0.0	+1.4/0.0	+19.0/0.0
	80.0/81.5(1.5)	82.0/81.5(−0.5)	50.5/46.5(−4.0)	3.0/3.0(0.0)	80.1/79.0(−1.1)	**69.5**/**70.5**(1.0)
	[−1.0, 4.0]	[−3.0, 2.0]	[−8.0, 0.0]	[0.0, 0.0]	[−3.9, 0.9]	[0.0, 2.5]
SMS	+3.0/0.0	−1.0/0.0	−8.0/0.0	0.0/0.0	−2.1/0.0	+2.0/0.0
Dense	79.0	78.0	54.0	1.0	81.4	72.0
Pair Skip	83.0	74.0	20.0	2.0	76.3	38.0

Table D8: Qwen L4 + L20: complete r32 transfer grid.

Source	ARC	PIQA	ASDiv	DART	ROPES	SMS
	79.5/**80.0**(0.5)	78.5/78.0(−0.5)	39.5/42.0(2.5)	1.5/1.0(−0.5)	77.4/77.3(−0.2)	31.5/43.0(11.5)
	[−2.0, 3.0]	[−2.5, 1.5]	[0.0, 5.5]	[−1.5, 0.0]	[−2.4, 1.9]	[7.0, 16.0]
ARC	0.0/ + 1.0	0.0/ − 1.0	0.0/ + 5.0	0.0/ − 1.0	0.0/ − 0.3	0.0/ + 23.0
	85.0/87.5(2.5)	**73.5**/**75.0**(1.5)	51.5/53.0(1.5)	3.0/3.5(0.5)	81.0/81.2(0.2)	74.0/69.5(−4.5)
	[0.5, 5.0]	[−3.0, 6.5]	[0.0, 3.5]	[0.0, 1.5]	[−2.1, 2.7]	[−7.5, −2.0]
PIQA	0.0/ + 5.0	0.0/ + 3.0	0.0/ + 3.0	0.0/ + 1.0	0.0/ + 0.3	0.0/ − 9.0
	61.0/68.0(7.0)	76.5/80.0(3.5)	**39.5**/**49.5**(10.0)	3.5/3.0(−0.5)	79.1/81.1(2.0)	63.0/68.0(5.0)
	[3.0, 11.5]	[−1.0, 8.0]	[4.5, 16.0]	[−1.5, 0.0]	[−2.1, 6.2]	[0.5, 10.0]
ASDiv	+5.0/ + 9.0	+4.0/ + 3.0	+9.0/ + 11.0	−1.0/0.0	+3.2/ + 0.8	+3.0/ + 7.0
	62.5/55.5(−7.0)	76.5/76.5(0.0)	27.5/17.5(−10.0)	**1.0**/**1.0**(0.0)	76.2/74.9(−1.4)	24.0/28.0(4.0)
	[−11.0, −3.5]	[−4.0, 4.5]	[−14.0, −6.5]	[0.0, 0.0]	[−3.3, 0.7]	[0.0, 8.5]
DART	+1.0/ − 15.0	−1.0/ + 1.0	0.0/ − 20.0	0.0/0.0	−0.5/ − 2.2	−1.0/ + 9.0
	87.0/85.0(−2.0)	82.5/77.5(−5.0)	10.5/16.0(5.5)	3.0/3.0(0.0)	**80.6**/**80.4**(−0.2)	84.0/69.5(−14.5)
	[−6.0, 2.5]	[−12.0, 2.0]	[1.5, 9.5]	[−2.5, 2.0]	[−5.1, 4.9]	[−20.0, −9.0]
ROPES	0.0/ − 4.0	−3.0/ − 7.0	−2.0/ + 13.0	−1.0/ + 1.0	−1.0/ + 0.6	+6.0/ − 35.0
	79.0/82.0(3.0)	78.0/78.0(0.0)	43.5/47.5(4.0)	4.5/4.0(−0.5)	76.4/77.1(0.7)	**70.5**/**70.5**(0.0)
	[1.0, 5.5]	[−2.5, 2.5]	[1.0, 7.0]	[−1.5, 0.0]	[−1.5, 2.8]	[−2.0, 2.0]
SMS	+6.0/0.0	0.0/0.0	+8.0/0.0	−1.0/0.0	+1.3/0.0	0.0/0.0
Dense	79.0	78.0	54.0	1.0	81.4	72.0
Pair Skip	83.0	74.0	20.0	2.0	76.3	38.0