KAT-PRUNE: STATE-ANCHORED PATH CORRECTION FOR TRAINING-FREE STRUCTURED LLM PRUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models achieve strong performance but incur high parameter and computational costs, hindering efficient deployment. Model compression is therefore essential, and structured pruning is attractive because it removes complete units such as attention heads and MLP channels to produce smaller dense models. Many training-free methods estimate structural importance under the dense model and retain these preferences as pruning changes the model, implicitly assuming they remain reliable after the state changes. Yet pruning alters hidden representations, unit interactions, and structural sensitivities, so a proxy measured at an earlier mask can lose fidelity to the current state. We term this self-induced surrogate mismatch. We introduce KAT-Prune (KL-guided Activation-ranked Transport Pruning). It revisits pruning decisions and previously removed structures at the current sparse state while preserving an exact physical parameter budget, and accepts partial corrections only when they reduce empirical teacher KL, without fine-tuning or post-pruning weight compensation. We establish the feasibility of its exact-budget correction path and derive a sufficient condition for a partial correction to decrease the population joint KL objective when its predicted gain exceeds the error from structural interactions. Across six LLMs and three pruning levels, KAT-Prune achieves the lowest WikiText-2 perplexity in 11 of 18 settings and the highest seven-task average accuracy in 12 of 18 among compared pruning methods. Relative to the strongest competing baseline in each setting, it reduces perplexity by up to 12.5% and improves seven-task average accuracy by up to 3.44 percentage points.

1 INTRODUCTION

Large language models (LLMs) exhibit strong capabilities in language understanding, generation, coding, and reasoning, yet their substantial memory and computational costs hinder efficient deployment. Model compression is therefore important for reducing these costs. Among existing compression approaches, structured pruning removes complete units such as attention heads and MLP channels, producing smaller dense models that can run without specialized sparse kernels. Its design centers on two related decisions: how to assess the importance of structural units and how to allocate the pruning budget.

Existing approaches address these two decisions from different angles. Earlier structured-pruning methods developed ways to identify removable structures and reduce the performance loss caused by their removal. For example, LLM-Pruner (Ma et al., 2023) assesses coupled structures using gradients, SliceGPT (Ashkboos et al., 2024) reshapes the representation space before structural removal, and FLAP (An et al., 2023) uses activation fluctuation for structural scoring and adaptive layer-wise allocation. For the score-based methods among them, structural scores and their normalization across layers can guide pruning and layer-wise allocation, but cannot directly measure the joint effect of removing multiple units under a fixed physical-parameter budget. More recent methods further examine these decisions: Týr-the-Pruner (Li et al., 2025) optimizes layer-wise sparsity distributions, PruneLLM (Yang et al., 2025) allocates pruning across modules based on their interactions, GRASPrune (Wang et al., 2026b) jointly optimizes FFN-channel and attention KV-head selection under a global budget, and GISP (Wang et al., 2026a) recomputes structural importance as sparsity increases. These methods advance budget allocation, modeling of structural interactions, and iterative assessment. Nevertheless, the relative importance of structures can still change as pruning

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.

proceeds, making the keep/remove choices in an existing sparse mask worth revisiting. Removing structural units alters the hidden states received by subsequent layers and the interactions among the remaining units, so a proxy measured at an earlier mask can become less reliable at the current one. We call this phenomenon self-induced surrogate mismatch.

To address this self-induced surrogate mismatch, we propose KAT-Prune (KL-guided Activation-ranked Transport Pruning). In detail, KAT-Prune starts from the dense model, constructs an initial sparse mask, and then remeasures candidate structures at each accepted sparse state. Conditional teacher KL, referenced to the dense model, quantifies the output distortion caused by candidate changes in that state, while a cost-aware dynamic program proposes a revised allocation at the same physical cost. Since these separate measurements do not account for all interactions among simultaneous changes, the proposal is not applied wholesale. Instead, KAT-Prune considers budget-neutral restore–delete exchanges and selects a partial update using joint teacher KL. Only an update that reduces empirical KL is accepted, after which the proxy is rebuilt at the new state. The pretrained weights remain frozen throughout.

Our contributions are:

- We identify and provide controlled evidence for self-induced surrogate mismatch in structured LLM pruning. Conditional-KL proxies exhibit high ranking fidelity at their anchor state but degrade when transferred across sparse states, showing that pruning preferences are inherently state dependent.
- We introduce KAT-Prune, which corrects pruning decisions through state-conditioned re-anchoring and exact-budget partial mask transport. A refreshed global allocation defines a correction direction, while budget-neutral restore–delete exchanges form a feasible path along which joint teacher KL selects a partial update. This allows previously removed structures to be reconsidered without modifying pretrained weights.
- We establish exact-budget path feasibility and derive a sufficient condition for joint KL descent under partial transport. Extensive experiments and ablations further evaluate the benefits of iterative re-anchoring, conditional-KL-guided allocation, and the resulting structured masks.

2 Related Work

LLM pruning and structured compression. Recent LLM pruning methods develop scalable criteria for identifying removable structures and allocating compression budgets across layers and modules. These methods exploit different signals: LLM-Pruner (Ma et al., 2023) uses gradient information for coupled structures; Wanda (Sun et al., 2024) combines weight magnitude with input-activation norms, with Wanda-SP (Sun et al., 2024) extending the score to structured groups; FLAP (An et al., 2023) relies on activation fluctuation together with adaptive layer-wise allocation and bias compensation; and SliceGPT (Ashkboos et al., 2024) reparameterizes the representation space before structural removal. Other methods mitigate pruning-induced errors through compensation: SlimGPT (Ling et al., 2024) updates retained weights using second-order information, RestoreLCC (Feng et al., 2025) restores lost components in attention activations, and RCPU (Haruta et al., 2026) uses rotation-constrained error compensation. These approaches differ in their scoring signals, pruning granularity, and recovery mechanisms, but they mainly focus on deciding which structures to remove or how to compensate for their removal. Once a sparse mask has been selected, previously committed within-layer pruning decisions are generally not explicitly reconsidered.

Global allocation and iterative structural search. Structural sensitivity varies across layers and structure types, motivating global rather than uniform pruning. FLAP (An et al., 2023) uses activation fluctuation for adaptive layer-wise allocation, while Týr-the-Pruner (Li et al., 2025) searches layer-wise pruning rates with a multi-rate supernet and evolutionary refinement. GRASPrune (Wang et al., 2026b) jointly allocates FFN channels and attention heads under a global budget, while GISP (Wang et al., 2026a) repeatedly recomputes structural importance as sparsity increases, producing nested subnetworks. Cooperative-game pruning (Ding et al., 2026) further models inter-layer dependencies, while TransAct (Shen et al., 2024) performs activation-guided iterative pruning. These studies show the value of global allocation and iterative refinement, but most retain earlier deletion decisions

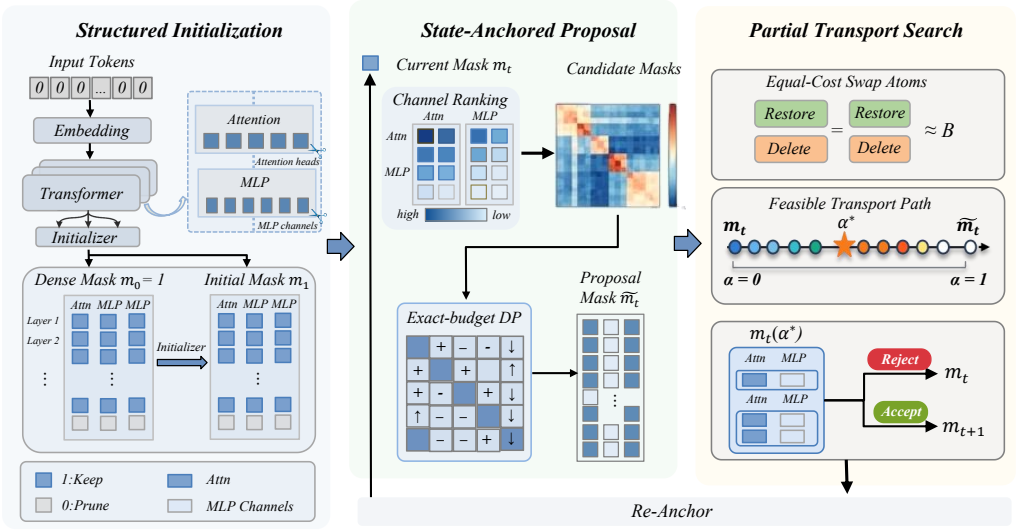


Figure 1: Activation rankings and conditional teacher KL first define an initial mask from the dense model, fixing the removed-parameter budget. At each correction round, refreshed measurements guide an exact-budget proposal, and budget-neutral restore–delete exchanges form a path toward it. Joint teacher KL on the shared calibration set selects a candidate only if it strictly improves upon the current mask. The accepted mask becomes the new anchor for rebuilding the surrogate. Throughout search, logical masks disable units while original weights remain frozen and available for restoration; physical removal occurs only after the final mask is selected.

while updating sparsity schedules, candidate configurations, or importance estimates. Our method instead reaches the target physical budget first, keeps it fixed, and revisits previous decisions through reversible, equal-cost restore–delete exchanges, allowing removed structures to be reconsidered as the sparse state evolves.

Sequential surrogate model management. Beyond pruning, SAIP-TR (Liang et al., 2020) solves discrete topology optimization for compliant mechanisms by iteratively constructing approximate integer-programming subproblems and restricting changes to discrete design variables with a trust region. Our method follows a related local-model principle in the discrete pruning space: a state-anchored additive KL surrogate proposes a correction direction, exact-budget transport defines feasible partial moves, and joint teacher KL selects the update before the surrogate is re-anchored at the new sparse state. In this way, local surrogate modeling is used not to make a one-shot pruning decision, but to support repeated, reversible correction under a fixed physical pruning budget.

3 Method

The proposed method adopts a state-aware structured pruning strategy, starting from the dense model to construct an initial sparse mask and fix its removed-parameter cost. It then reassesses attention heads and MLP channels in the current sparse model and proposes a revised allocation at the same cost. A partial restore–delete adjustment is accepted only if it reduces the output divergence from the dense model on the calibration set. Figure 1 summarizes the workflow from dense-model initialization to iterative mask correction.

3.1 Problem Formulation

Let $p_0(\cdot \mid x)$ denote the output distribution of the frozen dense teacher and $p_m(\cdot \mid x)$ that of the same pretrained model under a structural keep mask m, where $m_u = 1$ retains unit u and $m_u = 0$ disables it. The removable units $\mathcal{U}$ are MLP intermediate channels and legal attention units: one head in multi-head attention (MHA), or one KV head together with all associated query heads in grouped-query attention (GQA). Each unit has cost c_u, the number of parameters removed when it

is physically pruned. Let $\mathcal{L}_{\text{arch}}$ contain masks satisfying head-coupling constraints and retaining at least one unit in each structure group. Given an input distribution $\mathcal{D}$ and an attainable budget B, the objective is to preserve the teacher's output distribution while removing exactly B parameters:

$$m^\star \in \arg\min_{m \in \mathcal{L}_{\text{arch}}} \mathbb{E}_{x\sim\mathcal{D}} D_{\text{KL}}(p_0(\cdot \mid x) \| p_m(\cdot \mid x)) \quad \text{s.t.} \quad \sum_{u\in\mathcal{U}} c_u(1-m_u) = B. \tag{1}$$

Denote this expected KL by $F(m)$ and the removed parameter cost by $C(m) = \sum_{u\in\mathcal{U}} c_u(1-m_u)$. KL is computed over the full vocabulary and averaged over valid token positions, providing a label-free objective for mask selection.

The formulation assumes an attainable budget B. In practice, the nominal pruning target may not correspond to a valid structural mask. Initialization therefore starts from the dense mask $m_0 = \mathbf{1}$ and constructs a legal mask m_1. Its removed parameter cost, $B = C(m_1)$, becomes the fixed budget for all subsequent corrections, which remain in

$$\mathcal{M}_B = \{m \in \mathcal{L}_{\text{arch}} : C(m) = B\}. \tag{2}$$

To keep correction reversible, search uses logical masks to disable units while retaining the original frozen weight tensors, so masked units remain available for restoration. Physical pruning is performed only for final-model evaluation or deployment.

3.2 State-Conditioned Re-anchoring

A useful allocation surrogate must be conditioned on the sparse state in which candidate structural changes are evaluated. Upstream masking changes the activations entering later layers and, consequently, the effects of their structural units. Measurements from an earlier state may therefore no longer describe the current model. To re-anchor the surrogate, the method refreshes activation rankings and conditional teacher KL at each accepted mask m_t.

Activation rankings first define candidates within each layer–type group $g = (\ell, k)$. For unit i, let $h^k_{\ell i}(j; m_t)$ be its activation before the local structural mask. Its score is the mean squared activation over valid token positions and activation coordinates:

$$s^{(t)}_{\ell k i} = \frac{1}{N_t d_{\ell k}} \sum_{j=1}^{N_t} \|h^k_{\ell i}(j; m_t)\|_2^2. \tag{3}$$

Here, N_t counts valid token positions in the fixed shared calibration set $\mathcal{D}_{\text{cal}}$, and $d_{\ell k}$ is the activation dimension. The measured tensor is an intermediate channel before the MLP down projection or a head-output block before the attention output projection; under GQA, the associated query-head outputs are concatenated for each legal unit. Since measurement precedes the local structural mask while upstream masks remain active, disabled units can also be ranked in the current sparse context and reconsidered for restoration.

For each group g, let $\mathcal{R}_g$ be a finite set of legal deletion rates. Within each group g, increasing the deletion rate $r \in \mathcal{R}_g$ disables progressively more low-ranked units, producing nested candidate masks $M_{g,t}(r)$. To measure each candidate's effect on the model output, only group g is replaced while all other groups retain the current mask:

$$T_{g,t}(r) = \widehat{\mathbb{E}}_{x\in\mathcal{D}_{\text{cal}}} D_{\text{KL}}\big(p_0(\cdot \mid x) \| p_{m_t \leftarrow M_{g,t}(r)}(\cdot \mid x)\big)\,. \tag{4}$$

To combine these one-group measurements, define the population additive model at the current mask:

$$\bar{F}_t(m) = F(m_t) + \sum_{g=1}^{G} [F(m_t \leftarrow m_g) - F(m_t)]. \tag{5}$$

Here, G is the number of layer–type groups and m_g is group g of candidate m. The model starts from the current KL and adds each group's independently measured change. For tabulated candidates, its empirical counterpart differs from $S_t(\mathbf{r}) = \sum_g T_{g,t}(r_g)$ only by an allocation-independent constant. Cost-aware dynamic programming therefore allocates deletion rates by minimizing S_t over the tabulated candidates and their cost representation; activation-ranked boundary repair then enforces the actual physical budget. The resulting proposal $\widetilde{m}_t$ is retained only if it satisfies structural constraints and $C(\widetilde{m}_t) = B$ (Appendix A.1).

The anchored model is exact at m_t and under any single-group replacement, while simultaneous replacements introduce interactions not represented by the sum. This discrepancy is captured by the residual:

$$I_t(m) = F(m) - \bar{F}_t(m). \tag{6}$$

Re-anchoring restores single-group consistency at each accepted mask, but does not remove joint interactions. The proposal therefore guides a partial correction, whose extent is selected by joint teacher KL along an exact-budget path.

3.3 Exact-Budget Partial Mask Transport

The current mask and proposal have the same removed cost, so the units on which they differ have equal total restoration and deletion costs. To make the proposed correction reversible and budget preserving, the construction groups this difference into disjoint restore–delete atoms $e_j = (R_j, P_j)$. Each atom restores disabled units R_j and disables retained units P_j with equal physical cost:

$$w_j := \sum_{u \in R_j} c_u = \sum_{v \in P_j} c_v. \tag{7}$$

Atoms prioritize high-activation restorations and low-activation deletions using refreshed within-group percentiles. Exchanges are bundled when necessary so that every exposed prefix respects head coupling and retains nonempty groups (Appendix A.1). Applying the first q atoms defines

$$m_t^{(q)} = \mathrm{Apply}(m_t; e_1, \ldots, e_q), \qquad m_t^{(0)} = m_t, \quad m_t^{(J)} = \widetilde{m}_t, \quad m_t^{(q)} \in \mathcal{M}_B. \tag{8}$$

where J is the number of atoms. Equal restored and deleted costs preserve B at every prefix. A transport fraction $\alpha \in [0, 1]$ maps to the executable prefix $m_t(\alpha)$ whose cumulative cost is closest to $\alpha \sum_{j=1}^{J} w_j$; the endpoints retain the current mask or apply the full proposal.

The feasible path casts selection of the correction extent as a one-dimensional search over executable masks. The Screen stage evaluates each complete prefix mask using joint full-vocabulary teacher KL on the same fixed calibration set used to build the table. One-dimensional EGO-style Bayesian optimization (Jones et al., 1998) searches these executable prefixes (Appendix A.1). Let $f_t(\alpha) = \widehat{F}_{\mathrm{cal}}(m_t(\alpha))$ and $\mathcal{A}_t^{\mathrm{eval}}$ contain coordinates representing the current mask and all distinct evaluated candidates. Select $\widehat{\alpha}_t \in \arg\min_{\alpha \in \mathcal{A}_t^{\mathrm{eval}}} f_t(\alpha)$, favoring the current mask in a tie. The selected candidate is accepted only if $\widehat{\Delta}_t = f_t(\widehat{\alpha}_t) - f_t(0) < 0$; otherwise correction stops. Each KL evaluation measures the joint effect of all structural group changes present in that prefix mask.

Partial transport limits the extent of simultaneous changes. For $B > 0$ and $c_u > 0$, define the normalized distance $d_t(q) = B^{-1} \sum_{j \le q} w_j$. Under the smooth-gate regularity conditions in Appendix A.3.1, a finite cost-weighted cross-group curvature bound L_t gives

$$|I_t(m_t^{(q)})| \le \frac{L_t}{2} d_t(q)^2. \tag{9}$$

Theorem 1 (Local KL-descent condition)**.** *Under these regularity conditions, let $R_t(q) = F(m_t) - \bar{F}_t(m_t^{(q)})$ denote the population-surrogate improvement of a path prefix. If*

$$R_t(q) > \frac{L_t}{2} d_t(q)^2, \tag{10}$$

then the prefix preserves the exact budget and decreases the joint KL objective:

$$m_t^{(q)} \in \mathcal{M}_B, \qquad F(m_t^{(q)}) < F(m_t). \tag{11}$$

The bound explains the role of partial transport: a shorter step limits the interaction error associated with the additive surrogate. When the predicted gain exceeds this bound, the corresponding prefix decreases the population joint KL. Screen selects the implemented update by evaluating joint teacher KL along the feasible path on the fixed calibration set and accepting only a strict improvement over the current mask. The regularity conditions and proof are given in Appendix A.3.

Correction cycle. After acceptance, the updated mask becomes the new anchor: activations and conditional KL are remeasured to rebuild the surrogate. Correction stops if proposal construction or boundary repair fails, structural or exact-budget validation fails, no executable nonzero prefix exists, no strict empirical improvement is found, or the iteration limit is reached. Algorithm 1 summarizes initialization and correction.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

Models and evaluation. We evaluate Llama-2-7B, Llama-3-8B, Llama-3.1-8B, Llama-3.2-3B, Qwen2.5-14B, and Qwen3-8B at target pruning rates of 20%, 30%, and 50%. For KAT-Prune, initialization determines the attainable physical budget for each nominal target, and correction preserves that realized cost exactly. All performance evaluations use physically materialized models, with dense models serving as references. We report WikiText-2 (Merity et al., 2017) token perplexity (PPL) at sequence length 4096 and zero-shot accuracy on BoolQ (Clark et al., 2019), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2020), ARC-Easy and ARC-Challenge (Clark et al., 2018), OpenBookQA (Mihaylov et al., 2018), and PIQA (Bisk et al., 2020). Avg_7 is their unweighted mean, using normalized accuracy for HellaSwag, ARC-Easy, ARC-Challenge, OpenBookQA, and PIQA, and standard accuracy for BoolQ and WinoGrande.

Implementation details. KAT-Prune uses 128 nonoverlapping 1024-token blocks from the WikiText-2 training split, deterministically shuffled with seed 42. Activation scoring, conditional-KL table construction, Screen path search, mask acceptance, and monitoring all use these same blocks, which remain fixed across rounds. We allow at most eight iterations, including initialization and up to seven correction rounds.

Baseline results are obtained from the released implementations using the configurations recommended in the original papers. All methods use the same WikiText-2 calibration data: 128 nonoverlapping 1024-token blocks shuffled with seed 42. Each baseline retains its original scoring, search, reconstruction, and weight-update procedures. After pruning, the resulting models are physically materialized and evaluated under a unified setting.

4.2 MAIN RESULTS

We compare KAT-Prune with Wanda-SP (Sun et al., 2024), FLAP (An et al., 2023), Týr-the-Pruner (Li et al., 2025), and GRASPrune (Wang et al., 2026b) using our reproduced evaluations. The latter three incorporate bias compensation, weight adjustment, or learned rescaling; KAT-Prune preserves the pretrained values of all surviving weights. Table 1 reports results at target structured pruning rates of 20%, 30%, and 50%. KAT-Prune fixes the attainable budget produced by initialization and preserves that realized cost exactly throughout correction.

Table 1 shows that KAT-Prune achieves strong downstream accuracy across the evaluated model families. It obtains the highest Avg_7 in twelve of the eighteen model–pruning settings, including five of six models at both 20% and 30% target pruning. The gains are particularly pronounced at 30%: on Llama-3-8B and Llama-3.1-8B, KAT-Prune achieves Avg_7 scores of 57.14% and 54.34%, exceeding the strongest competing baseline in each setting by 3.44 and 1.26 percentage points, respectively. On Llama-2-7B, it achieves both the lowest PPL of 7.22 and the highest Avg_7 of 60.37%, with an accuracy gain of 2.80 percentage points over the strongest baseline. Taken together, KAT-Prune achieves competitive performance without post-pruning weight compensation.

KAT-Prune also achieves the lowest PPL in eleven of the eighteen settings, including Qwen2.5-14B at all three pruning levels and Llama-3-8B, Llama-3.1-8B, and Qwen3-8B at 50% target pruning. For example, on Qwen3-8B at 50%, it reduces PPL from 17.60 for the strongest competing baseline to 15.40, a relative reduction of 12.5%. Its downstream advantage is less consistent at this pruning level, where GRASPrune (Wang et al., 2026b) achieves higher Avg_7 on four of the six models. Thus, KAT-Prune's strongest overall results occur at moderate pruning levels, while its language-modeling performance remains competitive under more aggressive compression. At 30% pruning, KAT-Prune achieves the lowest PPL among the compared pruning methods on three of the six models, while

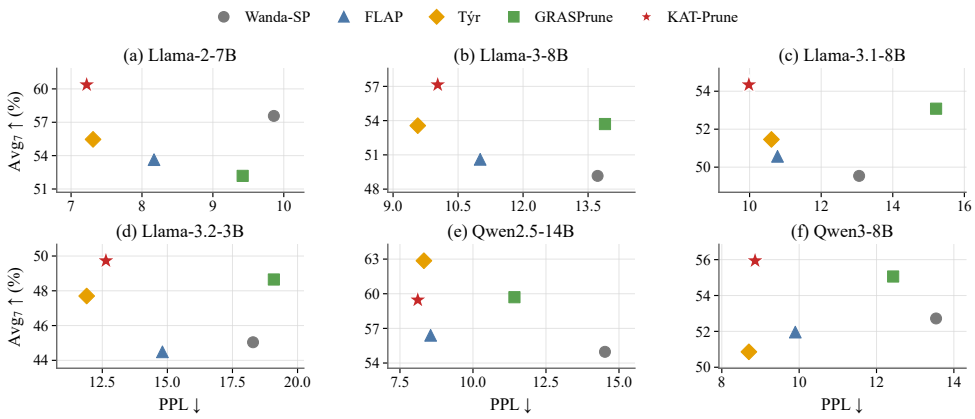


Figure 2: PPL–Avg_7 trade-off at 30% target pruning across six models. Lower PPL and higher Avg_7 are preferred; red stars denote KAT-Prune.

Table 1: WikiText-2 perplexity (PPL; ↓) and seven-task zero-shot average (Avg_7; ↑) at 20%, 30%, and 50% target structural pruning.

		20% target pruning		30% target pruning		50% target pruning	
Model	Method	PPL ↓	Avg_7 ↑	PPL ↓	Avg_7 ↑	PPL ↓	Avg_7 ↑
Llama-2-7B	Dense	5.12	66.89	5.12	66.89	5.12	66.89
	Wanda-SP	7.55	61.89	9.86	57.57	158.33	36.94
	FLAP	6.56	58.39	8.17	53.64	19.03	41.84
	Týr	6.25	43.33	7.31	55.46	**11.82**	42.87
	GRASPrune	7.64	58.41	9.42	52.18	21.49	44.50
	KAT-Prune	**6.15**	**62.48**	**7.22**	**60.37**	12.92	**46.51**
Llama-3-8B	Dense	5.76	70.68	5.76	70.68	5.76	70.68
	Wanda-SP	8.74	59.80	13.72	49.16	176.36	36.49
	FLAP	8.31	57.01	11.01	50.61	22.55	42.22
	Týr	8.13	59.26	**9.57**	53.56	17.34	41.48
	GRASPrune	11.11	59.76	13.89	53.70	28.80	**42.88**
	KAT-Prune	**7.86**	**60.44**	10.03	**57.14**	**16.65**	40.94
Llama-3.1-8B	Dense	5.84	71.43	5.84	71.43	5.84	71.43
	Wanda-SP	8.52	60.81	13.06	49.54	111.92	35.85
	FLAP	8.28	55.99	10.78	50.57	20.88	43.30
	Týr	9.13	59.20	10.61	51.46	18.24	42.68
	GRASPrune	10.82	58.24	15.21	53.08	24.79	**45.11**
	KAT-Prune	**7.83**	**62.36**	**9.98**	**54.34**	**16.50**	44.03
Llama-3.2-3B	Dense	7.29	65.28	7.29	65.28	7.29	65.28
	Wanda-SP	10.75	52.34	18.29	45.04	118.17	36.62
	FLAP	10.95	50.94	14.81	44.49	59.80	37.90
	Týr	**9.75**	**54.34**	**11.90**	47.70	**21.51**	39.49
	GRASPrune	13.14	52.70	19.09	48.65	32.87	41.12
	KAT-Prune	9.97	53.79	12.64	**49.73**	23.33	**42.03**
Qwen2.5-14B	Dense	4.93	72.71	4.93	72.71	4.93	72.71
	Wanda-SP	8.65	62.79	14.52	54.97	59.19	41.63
	FLAP	6.96	64.77	8.55	56.40	32.04	40.85
	Týr	9.10	53.40	8.32	**62.86**	17.09	44.30
	GRASPrune	9.26	65.52	11.42	59.70	22.92	**46.53**
	KAT-Prune	**6.41**	**67.64**	**8.11**	59.46	**16.16**	45.96
Qwen3-8B	Dense	6.52	70.62	6.52	70.62	6.52	70.62
	Wanda-SP	8.91	63.44	13.54	52.72	90.34	38.80
	FLAP	8.03	58.21	9.90	51.96	30.33	39.10
	Týr	**7.56**	62.83	**8.70**	50.86	17.60	40.60
	GRASPrune	10.03	58.84	12.43	55.06	19.40	**44.37**
	KAT-Prune	7.92	**64.22**	8.86	**55.94**	**15.40**	42.69

maintaining competitive Avg_7 performance. Figure 2 visualizes this PPL–Avg_7 trade-off across the six models.

Table 2 details the seven tasks at 30% pruning. On Llama-2-7B, KAT-Prune leads the pruned methods on all seven tasks; on Qwen3-8B, it leads on WinoGrande, ARC-E, and ARC-C. The corresponding Avg_7 gains over the strongest baselines are 2.80 and 0.88 percentage points. Appendix A.4.2 retains all models and pruning levels.

Table 2: Zero-shot downstream accuracy (%) for Llama-2-7B and Qwen3-8B at 30% target pruning.

Model	Method	BoolQ ↑	HellaSwag ↑	WinoGrande ↑	ARC-E ↑	ARC-C ↑	OBQA ↑	PIQA ↑	Avg_7 ↑
Llama-2-7B	Dense	79.45	76.26	69.38	75.59	44.71	44.20	78.62	66.89
	Wanda-SP	63.76	65.37	62.59	62.63	39.25	37.80	71.60	57.57
	FLAP	61.77	57.91	63.14	52.86	31.31	37.00	71.49	53.64
	Týr	66.39	62.70	60.14	57.87	33.79	36.20	71.11	55.46
	GRASPrune	51.99	60.61	59.19	55.43	33.62	34.60	69.80	52.18
	KAT-Prune	**68.56**	**68.47**	**66.77**	**64.35**	**39.68**	**39.60**	**75.14**	**60.37**
Qwen3-8B	Dense	82.87	78.56	72.53	81.65	56.83	42.20	79.71	70.62
	Wanda-SP	62.17	51.58	56.12	57.95	35.84	36.60	68.77	52.72
	FLAP	62.17	52.14	62.04	54.29	33.02	33.40	66.65	51.96
	Týr	38.47	58.18	61.33	56.65	36.52	**37.20**	67.68	50.86
	GRASPrune	**62.54**	**59.17**	58.88	60.86	37.20	37.00	**69.80**	55.06
	KAT-Prune	62.32	58.03	**62.43**	**64.35**	**39.08**	36.40	68.99	**55.94**

4.3 Ablation Studies

We ablate initialization, conditional-KL allocation, proxy refresh, and partial transport. *Initialization only* materializes the initial mask without correction. *Activation Z-score* replaces conditional-KL tables and DP with a global ranking of within-group standardized activation scores. *Frozen proxy* reuses the tables and DP target from the first correction. *Full exchange* applies each proposed transport in full, without joint-KL partial-step selection. All iterative variants preserve the initialized physical budget, use restore–delete transport, and share the terminal-mask reporting rule. Frozen proxy is evaluated at 30%; the other ablations use the same 30% target on both model families.

Table 3: Ablations at 30% pruning under identical removed cost; bold marks the best result in each setting.

	Llama-3-8B		Qwen3-8B	
Variant	PPL ↓	Avg_7 ↑	PPL ↓	Avg_7 ↑
Initialization only	29.43	44.66	9.34	55.53
Activation Z-score	73.62	43.45	17.02	47.01
Frozen proxy	10.49	**58.12**	9.03	54.97
Full exchange	18.77	52.62	17.08	47.15
KAT-Prune	**10.03**	57.14	**8.86**	**55.94**

Table 3 shows that KAT-Prune obtains the lowest PPL on both models. Relative to the initial mask, correction reduces PPL by 65.9% on Llama-3-8B and by 5.1% on Qwen3-8B. Activation Z-score yields 73.62 and 17.02 PPL, compared with 10.03 and 8.86 for KAT-Prune. Within the shared correction pipeline, conditional-KL tables and DP outperform activation-only ranking. Refreshing the proxy reduces PPL from 10.49 to 10.03 on Llama-3-8B and from 9.03 to 8.86 on Qwen3-8B, supporting state-conditioned remeasurement during iterative correction.

Correction raises Avg_7 from 44.66 to 57.14 on Llama-3-8B and from 55.53 to 55.94 on Qwen3-8B. Frozen proxy reaches 58.12 on Llama-3-8B, while proxy refresh produces the lower PPL on both models.

Under full exchange, PPL and Avg_7 fluctuate across rounds. At the terminal state, full exchange has higher PPL and lower Avg_7 than KAT-Prune on both models (Table 3). On Llama-3-8B, terminal PPL is 18.77 rather than 10.03; on Qwen3-8B, it is 17.08 rather than 8.86. Preserving the removed-parameter cost and constructing a complete transport path therefore do not, by themselves, reproduce the result obtained by selecting a partial update.

4.4 Correction Analysis and Inference Efficiency

The proxy-transfer experiment in Appendix A.2.3 tests whether a structural proxy remains reliable after the sparse mask changes. For each of Llama-2-7B and Llama-3.2-3B at 30% pruning, we use a set of 40 equal-cost exchanges that is feasible at both the iteration-1 and iteration-8 masks. We compare the ranking given by the iteration-1 proxy with joint ΔKL measured at both masks on a separate Probe split, then rebuild the proxy at iteration 8 and repeat the comparison on the same exchanges. Rebuilding the late proxy raises Spearman correlation from 0.526 to 0.894 on Llama-2-7B and from 0.706 to 0.952 on Llama-3.2-3B, supporting remeasurement at each accepted mask.

Fixed-budget trajectories compare the initial mask with the terminal mask without increasing the removed structural cost. Both PPL and Avg_7 improve from initialization to termination on Llama-2-7B and Llama-3.2-3B at 30% pruning (Figure 3).

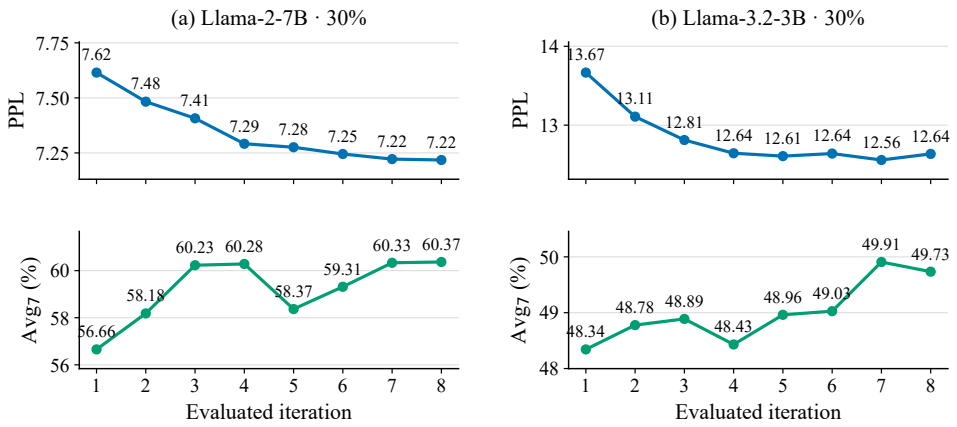


Figure 3: Optimization trajectories at 30% removal under fixed structural cost. In both models, terminal masks have lower PPL and higher Avg_7 than initial masks.

Compatibility with post-pruning reconstruction. With each final mask fixed, state-anchored reconstruction (SAR) uses activation second moments collected from the physically sparse model to estimate the contribution of removed inputs and folds that contribution into the retained attention-output and MLP down-projection weights. Applied after mask selection, SAR reduces PPL in all 15 settings of the five-model reconstruction study, with a mean relative reduction of 7.4% across the 15 settings. For Qwen3-8B at 50% pruning, for example, PPL decreases from 15.40 to 12.37, a 19.7% reduction. The closed-form update and complete measurements appear in Appendix A.4.1. Since SAR leaves the selected mask unchanged, its PPL reductions complement the gains from the preceding mask-correction procedure.

Search cost. For Llama-2-7B at 30% target pruning on one NVIDIA RTX 3090, a from-scratch run of initialization and seven correction rounds takes 62.67 GPU-hours in total, averaging 7.83 GPU-hours per iteration. Proposal construction, including the conditional-KL tables, accounts for 96.6% of this measured cost. This is the offline mask-search cost; the reported inference measurements concern the final physically materialized models.

Table 4: Measured inference efficiency of physically materialized KAT-Prune models.

Metric	20%	30%	50%
Backbone FLOPs reduction ↑	19.6%	29.4%	49.4%
Peak VRAM reduction ↑	17.4%	26.1%	43.5%
TTFT speedup ↑	1.06×	1.16×	1.43×
Prefill speedup ↑	1.11×	1.22×	1.59×
Decode speedup ↑	1.17×	1.25×	1.45×

Measured inference efficiency. Table 4 reports paired measurements on physically materialized models. At 30% and 50% removal, six-model mean backbone FLOPs fall by 29.4% and 49.4%, and peak VRAM by 26.1% and 43.5%, with faster inference.

5 Conclusion

KAT-Prune re-anchors conditional-KL measurements to evolving sparse states and revisits pruning decisions through exact-budget partial mask transport with frozen weights. Proxy-transfer comparisons show that an early-state proxy loses ranking fidelity when applied to a later mask, whereas re-anchoring improves agreement with measured joint KL. Under the fixed physical budget, the terminal masks have lower PPL and higher Avg_7 than the initial masks on Llama-2-7B and Llama-3.2-3B. In a separate two-model ablation, full exchange ends with higher PPL and lower Avg_7 than partial correction, consistent with a benefit from selecting partial updates. Across six LLMs and three pruning levels, it attains the lowest PPL in 11 of 18 settings and the highest seven-task accuracy in 12, with its strongest results at moderate pruning. At 50% removal, physically materialized models reduce six-model mean peak VRAM by 43.5% and achieve 1.43×, 1.59×, and 1.45× speedups in TTFT, prefill, and decode, respectively. Revisiting structural choices at the current sparse state can therefore improve the final model while preserving its physical pruning budget.

AI USE STATEMENT

We used generative AI tools to assist with polishing the manuscript, literature search, research ideation, and experiment execution. All AI-assisted contributions were reviewed by the authors.

REFERENCES

Yongqi An, Xu Zhao, Tao Yu, Ming Tang, and Jinqiao Wang. Fluctuation-based adaptive structured pruning for large language models, 2023. URL https://arxiv.org/abs/2312.11983.

Saleh Ashkboos, Maximilian Croci, Marcelo Gennari do Nascimento, Torsten Hoefler, and James Hensman. Slicegpt: Compress large language models by deleting rows and columns. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 11682–11701, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/316648eb8b4ffb6010f531b07848c300-Paper-Conference.pdf.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: reasoning about physical commonsense in natural language. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 7432–7439. AAAI Press, 2020. doi: 10.1609/AAAI.V34I05.6239. URL https://doi.org/10.1609/aaai.v34i05.6239.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300/.

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, 2018. URL https://arxiv.org/abs/1803.05457.

Xuan Ding, Pengyu Tong, Ranjie Duan, Yunjian Zhang, Rui Sun, and Yao Zhu. Pruning as a cooperative game: Surrogate-assisted layer contribution estimation for large language models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 79775–79799, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/8105a80cea247fa6f8441a8069b65f2a-Paper-Conference.pdf.

Zijian Feng, Hanzhang Zhou, Zixiao Zhu, Tianjiao Li, Chua Deryl, Lee Onn Mak, Gee Ng, and Kezhi Mao. Restoring pruned large language models via lost component compensation. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 6462–6492. Curran Associates, Inc., 2025. doi: 10.52202/085713-0225. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/099ba28b20462dac7bf057d21c27a011-Paper-Conference.pdf.

Shuichiro Haruta, Kazunori Matsumoto, Zhi Li, Yanan Wang, and Mori Kurokawa. RCPU: Rotation-constrained error compensation for structured pruning of large language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=t6xiPRvynD.

Babak Hassibi and David Stork. Second order derivatives for network pruning: Optimal brain surgeon. In S. Hanson, J. Cowan, and C. Giles (eds.), *Advances in Neural Information Processing Systems*, volume 5. Morgan-Kaufmann, 1992. URL https://proceedings.neurips.cc/paper_files/paper/1992/file/303ed4c69846ab36c2904d3ba8573050-Paper.pdf.

Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, December 1998. ISSN 1573-2916. doi: 10.1023/A:1008306431147. URL https://doi.org/10.1023/A:1008306431147.

Guanchen Li, Yixing Xu, Zeping Li, Ji Liu, Xuanwu Yin, Dong Li, and Emad Barsoum. Týr-the-pruner: Structural pruning LLMs via global sparsity distribution optimization. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 149163–149192. Curran Associates, Inc., 2025. doi: 10.52202/085713-4989. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/db2cbf43a349bc866111e791b58c7bf4-Paper-Conference.pdf.

Yuang Liang, Kai Sun, and GengDong Cheng. Discrete variable topology optimization for compliant mechanism design via sequential approximate integer programming with trust region (SAIP-TR). *Structural and Multidisciplinary Optimization*, 62(6):2851–2879, December 2020. ISSN 1615-1488. doi: 10.1007/s00158-020-02693-2. URL https://doi.org/10.1007/s00158-020-02693-2.

Gui Ling, Ziyang Wang, Yuliang Yan, and Qingwen Liu. SlimGPT: Layer-wise structured pruning for large language models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 107112–107137. Curran Associates, Inc., 2024. doi: 10.52202/079017-3401. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/c1c44e46358e0fb94dc94ec495a7fb1a-Paper-Conference.pdf.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. LLM-Pruner: On the structural pruning of large language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 21702–21720. Curran Associates, Inc., 2023. doi: 10.52202/075280-0950. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/44956951349095f74492a5471128a7e0-Paper-Conference.pdf.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=Byj72udxe.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2381–2391, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1260. URL https://aclanthology.org/D18-1260/.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 8732–8740. AAAI Press, 2020. doi: 10.1609/AAAI.V34I05.6399. URL https://doi.org/10.1609/aaai.v34i05.6399.

Bowen Shen, Zheng Lin, Daren Zha, Wei Liu, Jian Luan, Bin Wang, and Weiping Wang. Pruning large language models to intra-module low-rank architecture with transitional activations. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9781–9793, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.582. URL https://aclanthology.org/2024.findings-acl.582/.

Mingjie Sun, Zhuang Liu, Anna Bair, and Zico Kolter. A simple and effective pruning approach for large language models. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 4942–4964,

2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/14c856c7a41297804de4c4890e846b25-Paper-Conference.pdf.

Ziyan Wang, Enmao Diao, Qi Le, Pu Wang, Minwoo Lee, Shu-ping Yeh, Evgeny Stupachenko, Hao Feng, and Li Yang. From local to global: Revisiting structured pruning paradigms for large language models. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 35720–35739, San Diego, California, United States, July 2026a. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.1653. URL https://aclanthology.org/2026.acl-long.1653/.

Ziyang Wang, Jiangfeng Xiao, Chuan Xiao, Ruoxiang LI, Rui Mao, and Jianbin Qin. GRASPrune: Global gating for budgeted structured pruning of large language models. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10719–10736, San Diego, California, United States, July 2026b. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.491. URL https://aclanthology.org/2026.acl-long.491/.

Mingzhe Yang, Sihao Lin, Changlin Li, and Xiaojun Chang. Let LLM tell what to prune and how much to prune. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 70833–70849. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/yang25p.html.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

A Appendix

A.1 Algorithmic Details

A.1.1 Allocation and Budget Repair

For a group with n_g removable units, the nominal deletion-rate grid $\{0, 0.05, \ldots, 1\}$ maps rate r to $\min\{\mathrm{round}(rn_g), n_g - 1\}$ deletions. The resulting group-specific grid is denoted by $\mathcal{R}_g$.

For a mask assembled from tabulated candidates, $m_g = M_{g,t}(r_g)$, the empirical anchored model satisfies

$$\widehat{\widehat{F}}_t(m) = \widehat{F}(m_t) + \sum_{g=1}^{G} [T_{g,t}(r_g) - \widehat{F}(m_t)] = S_t(\mathbf{r}) - (G-1)\widehat{F}(m_t). \tag{12}$$

Because the final term is constant within a round, minimizing S_t gives the same ordering over the tabulated allocations as the empirical anchored model.

Let $C_g(r)$ be the removed physical cost of candidate $M_{g,t}(r)$, $C_{\mathrm{tot}} = \sum_{u \in \mathcal{U}} c_u$, and $B_t^{\mathrm{ref}} = \mathrm{round}(\rho C_{\mathrm{tot}})$ at initialization or B thereafter. With $\kappa = \gcd\{c_u : u \in \mathcal{U}\}$ and $\widehat{C}_g(r) = C_g(r)/\kappa$, allocation uses

$$D_t(g, b) = \min_{r \in \mathcal{R}_g : \widehat{C}_g(r) \le b} \left[D_t(g-1, b - \widehat{C}_g(r)) + T_{g,t}(r) \right], \qquad D_t(0,0) = 0. \tag{13}$$

Unreachable states have value $+\infty$. The terminal state is chosen first by proximity to $\mathrm{round}(B_t^{\mathrm{ref}}/\kappa)$ and then by table objective.

Boundary repair refines the physical cost after this grid-level allocation. Following the refreshed activation order, it applies deletions or restorations that reduce the residual cost gap subject to the structural constraints; moves with the same effect are ordered by activation score per unit cost. The repaired mask is treated as a complete candidate and evaluated jointly during transport. At initialization, the first legal repaired mask sets the fixed budget $B = C(m_1)$. In subsequent rounds, a proposal enters transport only if $\widetilde{m}_t \in \mathcal{L}_{\mathrm{arch}}$ and $C(\widetilde{m}_t) = B$; otherwise, correction terminates at the current mask.

A.1.2 Cost-Neutral Atomization and Search Schedule

Let $\mathcal{R}_t$ and $\mathcal{P}_t$ be the units restored and newly deleted between m_t and $\widetilde{m}_t$. The evaluated architectures have two uniform unit-cost classes: an MLP channel and a legal MHA/GQA attention unit. For costs c_1, c_2, same-type changes are paired first; remaining cross-type changes form minimum cost-neutral packets containing c_2/g units of type 1 and c_1/g units of type 2, where $g = \gcd(c_1, c_2)$.

Restorations receive priority π_u and deletions $1 - \pi_u$, where π_u is the refreshed within-type activation percentile. Units are sorted by decreasing priority within each type–action list, and atoms are ordered by the minimum and then mean priority of their constituents. Adjacent atoms are bundled only when required for an architecture-legal prefix.

The six warm-start fractions are

$$\mathcal{A}_0 = \{1/32, 1/16, 1/8, 1/4, 1/2, 1\}. \tag{14}$$

Each fraction maps to the nearest cumulative atom cost, with repeated prefixes evaluated once. A one-dimensional Gaussian-process optimizer with a Matérn-$5/2$ kernel and expected improvement evaluates at most 14 additional nonzero prefixes. Thus each round evaluates at most 20 distinct nonzero prefixes before applying the joint-KL selection and acceptance rule in Section 3.

A.1.3 Complete Procedure

The allocation, budget-repair, and transport components described above form an iterative correction procedure under a fixed realized budget. Initialization constructs the first legal mask and fixes its removed cost, after which each round refreshes the state-conditioned surrogate, constructs an exact-budget proposal, and evaluates feasible transport prefixes by joint Screen KL. The procedure accepts only a strict improvement and returns the current mask when proposal construction, path construction, or candidate selection yields no valid update. Algorithm 1 gives the complete procedure.

Algorithm 1 KAT-Prune

Require: Frozen teacher p_0, target ratio ρ, shared calibration set $\mathcal{D}_{\text{cal}}$, maximum iterations T (including initialization)

At m_0, build the activation–KL surrogate; construct a valid m_1 and fix $B = C(m_1)$
$m_{\text{current}} \leftarrow m_1$
for $t = 1, \ldots, T-1$ **do**
  $m_t \leftarrow m_{\text{current}}$
  Refresh activations and conditional KL table at m_t
  Construct and validate an exact-budget proposal $\widetilde{m}_t$
  if construction fails or $\widetilde{m}_t \notin \mathcal{M}_B$ **then**
    break
  end if
  Build activation-priority-ordered cost-neutral transport prefixes
  if there is no nonzero prefix **then**
    break
  end if
  Evaluate the current mask and distinct prefixes from the six warm-start fractions on Screen
  Query at most 14 additional unevaluated nonzero prefixes by BO
  Select $\widehat{\alpha}_t$ by minimum Screen KL, favoring the current mask in a tie
  if $m_t(\widehat{\alpha}_t) = m_t$ **then**
    break
  end if
  Compute $\widehat{\Delta}_t = f_t(\widehat{\alpha}_t) - f_t(0)$ from the Screen evaluations
  if $\widehat{\Delta}_t \geq 0$ **then**
    break
  end if
  $m_{t+1} \leftarrow m_t(\widehat{\alpha}_t)$; $m_{\text{current}} \leftarrow m_{t+1}$
end for
return m_{current}

A.2 Experimental Protocols

A.2.1 Activation Z-score Ablation

For the ablation defined in Section 4.3, the score is $z_{g,u}^{(t)} = (s_{g,u}^{(t)} - \mu_{g,t})/(\sigma_{g,t} + 10^{-8})$, with all scores set to zero when $\sigma_{g,t} = 0$. Eligible units are ranked globally in ascending z. Exact-budget repair enumerates feasible attention deletion counts and their corresponding MLP counts, then minimizes the cost change from the ranked prefix, followed by the number of changed units and the attention count. Each run preserves the physical budget attained by its initial mask.

A.2.2 Frozen-Proxy Ablation

The frozen-proxy variant starts from the same initialization as KAT-Prune and constructs the conditional-KL tables and DP target for the first correction. Subsequent rounds keep this target fixed while rebuilding the exact-budget restore–delete path from the current mask and applying the same joint-KL screen and stopping rule. This comparison isolates proxy refresh while retaining the transport and acceptance mechanisms.

A.2.3 State-Anchored Proxy-Fidelity Protocol

For Llama-2-7B and Llama-3.2-3B at 30% removal, the study compares the mask at iteration 1 with the iteration-8 mask. Each model uses the same 40 exchanges at both anchors: ten MLP-to-MLP, ten attention-to-attention, and ten in each cross-type direction. All exchanges have the same one-sided transport cost and are feasible at both masks.

The early proxy is compared with joint ΔKL at both anchors, while a proxy rebuilt at iteration 8 is compared with the late joint effects. Each proxy sums the conditional-KL changes of the modified groups. Joint ΔKL is measured after applying the complete exchange on a disjoint Probe split

comprising WikiText-2 training blocks $[1152, 1280)$. Probe is not used to construct candidates. Pairwise accuracy is the fraction of exchange pairs ordered consistently by proxy and joint effect.

Table 5: Proxy fidelity at 30% removal. Early current compares the early proxy with early joint ΔKL; late stale compares that proxy with late joint ΔKL; late current rebuilds the proxy at the late mask. Higher is better; bold entries use state-matched proxies.

Model	Metric	Early current	Late stale	Late current
Llama-2-7B	Spearman $\rho \uparrow$	**0.908**	0.526	**0.894**
	Pairwise accuracy $\uparrow$	**0.886**	0.685	**0.883**
Llama-3.2-3B	Spearman $\rho \uparrow$	**0.907**	0.706	**0.952**
	Pairwise accuracy $\uparrow$	**0.897**	0.782	**0.921**

At the late mask, rebuilding the proxy raises Spearman correlation from 0.526 to 0.894 for Llama-2-7B and from 0.706 to 0.952 for Llama-3.2-3B. Pairwise accuracy increases from 0.685 to 0.883 and from 0.782 to 0.921, respectively. The fixed candidate sets attribute these gains to state-conditioned measurement.

A.3 Theoretical Analysis

A.3.1 Smooth-Gate Basis for the Interaction Bound

Assign each removable unit a scalar gate $z_u \in [0, 1]$, shared across the activation coordinates coupled within that unit. Gates multiply the same pre-mask activations as the logical masks, with all pretrained tensors and architectural dimensions fixed. For every legal binary mask m,

$$f(z) := \mathbb{E}_{x\sim\mathcal{D}} D_{\mathrm{KL}}(p_0(\cdot \mid x) \| p_z(\cdot \mid x)), \qquad f(m) = F(m). \tag{15}$$

We assume bounded context length, finite vocabulary, and twice continuously differentiable forward maps on a neighborhood of the gate cube. Standard Transformer operations satisfy this condition in exact arithmetic, with the fixed positive ε in RMSNorm.

For the current mask and proposal, define the compact path domain

$$\mathcal{K}_t = \prod_{u\in\mathcal{U}} [\min(m_{t,u}, \widetilde{m}_{t,u}), \max(m_{t,u}, \widetilde{m}_{t,u})]. \tag{16}$$

Every prefix and every hybrid formed by taking a changed group from either endpoint lies in $\mathcal{K}_t$. For a prefix m, let $\mathcal{G}_t(m) = \{g : m_g \neq (m_t)_g\}$ and let z_A take group g from m when $g \in A \subseteq \mathcal{G}_t(m)$ and from m_t otherwise. Such binary hybrids remain architecture legal under the group-local constraints but need not satisfy the physical budget; the implemented path contains legal, exact-budget masks.

Set $\omega_u = c_u/(2B) > 0$. Compactness and smoothness give the finite constant

$$L_t = \max_{\substack{g\neq h \\ u\in\mathcal{U}_g,\ v\in\mathcal{U}_h}} \sup_{z\in\mathcal{K}_t} \frac{|\partial^2_{uv} f(z)|}{\omega_u \omega_v} < \infty, \tag{17}$$

with $L_t = 0$ when no cross-group pair exists. Hence, for vectors supported on distinct groups,

$$|v_g^\top \nabla^2_{gh} f(z) v_h| \le L_t \left(\sum_{u\in\mathcal{U}_g} \omega_u |v_{g,u}| \right) \left(\sum_{v\in\mathcal{U}_h} \omega_v |v_{h,v}| \right). \tag{18}$$

Lemma 1 (Smooth gates imply bounded discrete interactions)**.** *For a path prefix m, distinct changed groups g, h, and $A \subseteq \mathcal{G}_t(m) \setminus \{g, h\}$,*

$$|\Delta_h \Delta_g f(z_A)| \le L_t a_{t,g}(m) a_{t,h}(m), \qquad a_{t,g}(m) = \frac{1}{2B} \sum_{u\in\mathcal{U}_g} c_u |m_u - m_{t,u}|. \tag{19}$$

Proof of Lemma 1. Let δ_g embed $m_g - (m_t)_g$ in the full gate vector. The discrete mixed difference is

$$\Delta_h \Delta_g f(z_A) = f(z_{A\cup\{g,h\}}) - f(z_{A\cup\{h\}}) - f(z_{A\cup\{g\}}) + f(z_A). \tag{20}$$

Applying the fundamental theorem of calculus along δ_g and δ_h gives

$$\Delta_h \Delta_g f(z_A) = \int_0^1 \int_0^1 \delta_g^\top \nabla^2_{gh} f(z_A + s\delta_g + r\delta_h)\delta_h \, dr \, ds. \tag{21}$$

All integrand points lie in $\mathcal{K}_t$, and

$$a_{t,g}(m) = \sum_{u \in \mathcal{U}_g} \omega_u |\delta_{g,u}| = \frac{1}{2B} \sum_{u \in \mathcal{U}_g} c_u |m_u - m_{t,u}|. \tag{22}$$

Equation (18) therefore yields

$$|\Delta_h \Delta_g f(z_A)| \le L_t a_{t,g}(m) a_{t,h}(m). \tag{23}$$

A.3.2 Anchored Residual Bound

Proposition 1 (Quadratic anchored residual)**.** *Under Lemma 1, every transport prefix satisfies the residual bound in Eq.* (9)*.*

Proof of Proposition 1. Order the changed groups of a prefix mask m as $g_1, \dots, g_K$, and let $z^{(j)} = z_{\{g_1,\dots,g_j\}}$. Every intermediate mask is therefore in the hybrid family covered by Lemma 1. Telescoping gives

$$F(m) - F(m_t) = \sum_{j=1}^{K} \Delta_{g_j} f(z^{(j-1)}), \tag{24}$$

where $\Delta_g f(z) = f(z \leftarrow m_g) - f(z)$. Equation (5) gives $\bar{F}_t(m) - F(m_t) = \sum_j \Delta_{g_j} f(m_t)$. For each j, telescope the change of this marginal over the preceding group replacements. Applying Eq. (23) to every mixed difference yields

$$\begin{aligned} |I_t(m)| &\le L_t \sum_{j=1}^{K} a_{t,g_j}(m) \sum_{i<j} a_{t,g_i}(m) \\ &= \frac{L_t}{2} \left[\left(\sum_g a_{t,g}(m) \right)^2 - \sum_g a_{t,g}(m)^2 \right]. \end{aligned} \tag{25}$$

For a cost-neutral prefix, the weighted symmetric difference from m_t contains equal restored and deleted cost, so $\sum_g a_{t,g}(m) = d_t(q)$. Dropping the nonnegative term $\sum_g a_{t,g}(m)^2$ and substituting this identity proves Eq. (9).

A.3.3 Proof of Theorem 1

Structural legality follows from the atom construction: only simultaneous exchanges whose exposed prefix remains in $\mathcal{L}_{\text{arch}}$ are admitted, with adjacent neutral exchanges bundled when necessary. Separately, every atom changes removed cost by $\sum_{v \in P_j} c_v - \sum_{u \in R_j} c_u = 0$, so $C(m_t^{(q)}) = B$ for every prefix by induction. Hence $m_t^{(q)} \in \mathcal{M}_B$. Moreover,

$$\begin{aligned} F(m_t^{(q)}) - F(m_t) &= -R_t(q) + I_t(m_t^{(q)}) \\ &\le -R_t(q) + \frac{L_t}{2} d_t(q)^2 < 0, \end{aligned} \tag{26}$$

where the strict inequality follows from Eq. (10). When $L_t = 0$, the condition reduces to $R_t(q) > 0$.

A.4 Additional Results

A.4.1 Post-Pruning Reconstruction Results

For retained indices K and removed indices R, SAR uses the damped OBC-style coefficient (Hassibi & Stork, 1992)

$$Q = (H_{KK} + \lambda I)^{-1} H_{KR} \tag{27}$$

and updates the retained output-projection weights as $W'_K = W_K + W_R Q^\top$. We set λ to 10^{-2} times the absolute mean diagonal of H_{KK}, lower-bounded by 10^{-12} before multiplication. The second moments are collected by forward passes through the physically sparse prefix using 128 sequences of length 1024. The update is applied to attention output and MLP down projections without gradients or fine-tuning.

Table 6 reports PPL before and after reconstruction for all 15 settings in the five-model reconstruction study.

Table 6: Complete post-pruning reconstruction results. KAT-Prune uses the final mask without weight reconstruction; KAT-Prune+SAR applies state-anchored closed-form reconstruction to the same mask.

		PPL ↓	
Model	Target	KAT	+SAR
Llama-2-7B	20%	6.15	**6.00**
	30%	7.22	**6.88**
	50%	12.92	**11.39**
Llama-3-8B	20%	7.86	**7.56**
	30%	10.03	**9.35**
	50%	16.65	**15.71**
Llama-3.1-8B	20%	7.83	**7.67**
	30%	9.98	**9.16**
	50%	16.50	**15.65**
Llama-3.2-3B	20%	9.97	**9.45**
	30%	12.64	**11.51**
	50%	23.33	**20.24**
Qwen3-8B	20%	7.92	**7.51**
	30%	8.86	**8.13**
	50%	15.40	**12.37**

A.4.2 Detailed Downstream Results

Tables 7–9 report zero-shot accuracy on all seven downstream tasks at 20%, 30%, and 50% target pruning, respectively. Avg_7 uses the same metric definitions as Table 1.

Table 7: Zero-shot downstream accuracy (%) at 20% target pruning.

Model	Method	BoolQ ↑	HellaSwag ↑	WinoGrande ↑	ARC-E ↑	ARC-C ↑	OBQA ↑	PIQA ↑	Avg_7 ↑
Llama-2-7B	Dense	79.45	76.26	69.38	75.59	44.71	44.20	78.62	66.89
	Wanda-SP	66.73	71.83	66.30	68.22	41.64	40.80	77.69	61.89
	FLAP	67.16	65.71	65.75	60.73	36.18	38.80	74.43	58.39
	Týr	45.75	53.56	48.62	38.47	28.16	26.00	62.79	43.33
	GRASPrune	**68.38**	69.22	63.06	57.37	35.15	**41.60**	74.10	58.41
	KAT-Prune	64.71	**72.60**	**67.72**	**69.49**	**43.43**	**41.60**	**77.80**	**62.48**
Llama-3-8B	Dense	81.83	79.22	73.09	80.72	54.18	45.00	80.74	70.68
	Wanda-SP	66.51	64.12	**67.80**	65.32	41.21	38.60	75.03	59.80
	FLAP	**72.20**	57.66	64.48	58.16	34.73	39.40	72.42	57.01
	Týr	61.74	64.78	66.69	**66.25**	39.08	41.00	75.30	59.26
	GRASPrune	63.91	**69.57**	65.19	62.29	39.93	**41.80**	**75.63**	59.76
	KAT-Prune	63.61	**69.57**	67.64	65.19	**41.47**	40.60	75.03	**60.44**
Llama-3.1-8B	Dense	83.00	79.34	74.59	82.11	54.86	45.20	80.90	71.43
	Wanda-SP	72.32	64.77	67.32	65.53	42.06	38.60	75.03	60.81
	FLAP	68.29	58.09	64.25	57.20	34.22	37.80	72.09	55.99
	Týr	61.56	62.79	65.51	**68.77**	**42.49**	39.00	74.27	59.20
	GRASPrune	66.39	67.55	62.75	60.27	37.63	38.80	74.27	58.24
	KAT-Prune	**74.74**	**70.05**	**67.72**	68.14	41.38	**39.40**	**75.08**	**62.36**
Llama-3.2-3B	Dense	73.79	74.08	69.14	74.62	46.25	41.00	78.07	65.28
	Wanda-SP	52.78	54.83	58.17	61.49	34.13	34.40	70.57	52.34
	FLAP	61.10	50.59	57.77	54.46	30.03	33.60	68.99	50.94
	Týr	**62.69**	55.41	59.59	**61.95**	34.39	36.20	70.18	**54.34**
	GRASPrune	59.11	57.84	58.17	57.11	31.40	34.00	71.27	52.70
	KAT-Prune	50.34	**61.19**	**61.80**	59.47	**34.47**	**37.00**	**72.25**	53.79
Qwen3-8B	Dense	82.87	78.56	72.53	81.65	56.83	42.20	79.71	70.62
	Wanda-SP	**77.34**	65.90	63.22	73.57	47.87	41.40	74.76	63.44
	FLAP	62.26	62.67	68.11	62.21	41.38	39.40	71.44	58.21
	Týr	72.14	**71.64**	69.22	65.87	45.48	40.00	75.46	62.83
	GRASPrune	61.19	66.70	62.35	62.96	42.32	**42.60**	73.78	58.84
	KAT-Prune	69.36	68.17	**69.46**	**76.43**	**48.98**	41.40	**75.73**	**64.22**
Qwen2.5-14B	Dense	85.26	82.92	75.30	79.00	58.45	45.60	82.43	72.71
	Wanda-SP	66.85	69.88	66.38	72.85	50.43	39.20	73.94	62.79
	FLAP	**82.63**	65.97	67.56	75.21	47.27	43.20	71.55	64.77
	Týr	73.98	70.00	69.93	37.75	24.57	40.80	56.80	53.40
	GRASPrune	76.02	**75.78**	68.03	69.53	49.91	42.40	**76.93**	65.52
	KAT-Prune	79.45	73.55	**70.72**	**77.57**	**50.77**	**44.60**	76.82	**67.64**

Table 8: Zero-shot downstream accuracy (%) at 30% target pruning.

Model	Method	BoolQ ↑	HellaSwag ↑	WinoGrande ↑	ARC-E ↑	ARC-C ↑	OBQA ↑	PIQA ↑	Avg_7 ↑
Llama-2-7B	Dense	79.45	76.26	69.38	75.59	44.71	44.20	78.62	66.89
	Wanda-SP	63.76	65.37	62.59	62.63	39.25	37.80	71.60	57.57
	FLAP	61.77	57.91	63.14	52.86	31.31	37.00	71.49	53.64
	Týr	66.39	62.70	60.14	57.87	33.79	36.20	71.11	55.46
	GRASPrune	51.99	60.61	59.19	55.43	33.62	34.60	69.80	52.18
	KAT-Prune	**68.56**	**68.47**	**66.77**	**64.35**	**39.68**	**39.60**	**75.14**	**60.37**
Llama-3-8B	Dense	81.83	79.22	73.09	80.72	54.18	45.00	80.74	70.68
	Wanda-SP	50.21	47.82	57.70	54.12	31.91	34.40	67.95	49.16
	FLAP	68.26	47.65	59.67	48.11	29.52	35.00	66.05	50.61
	Týr	62.05	54.92	64.40	54.59	31.57	**37.20**	70.18	53.56
	GRASPrune	64.01	59.97	58.80	55.35	33.87	34.40	69.53	53.70
	KAT-Prune	**73.27**	**60.65**	**66.54**	**56.40**	**35.49**	36.40	**71.22**	**57.14**
Llama-3.1-8B	Dense	83.00	79.34	74.59	82.11	54.86	45.20	80.90	71.43
	Wanda-SP	50.12	48.81	56.27	54.67	32.08	**37.00**	67.85	49.54
	FLAP	**69.20**	48.17	60.62	48.23	27.56	33.80	66.43	50.57
	Týr	51.80	54.55	60.14	**55.72**	32.59	35.20	70.24	51.46
	GRASPrune	63.36	**59.57**	58.25	52.65	**33.70**	34.20	69.86	53.08
	KAT-Prune	68.32	58.33	**64.17**	52.65	31.83	34.60	**70.51**	**54.34**
Llama-3.2-3B	Dense	73.79	74.08	69.14	74.62	46.25	41.00	78.07	65.28
	Wanda-SP	47.92	41.23	53.12	51.26	27.73	30.60	63.44	45.04
	FLAP	48.99	42.99	55.17	43.69	25.94	31.80	62.84	44.49
	Týr	49.91	45.42	**58.88**	51.18	28.84	33.60	66.05	47.70
	GRASPrune	**61.10**	45.59	53.83	50.00	**30.80**	33.40	65.83	48.65
	KAT-Prune	49.91	**52.72**	58.09	**54.25**	30.55	**34.40**	**68.23**	**49.73**
Qwen3-8B	Dense	82.87	78.56	72.53	81.65	56.83	42.20	79.71	70.62
	Wanda-SP	62.17	51.58	56.12	57.95	35.84	36.60	68.77	52.72
	FLAP	62.17	52.14	62.04	54.29	33.02	33.40	66.65	51.96
	Týr	38.47	58.18	61.33	56.65	36.52	**37.20**	67.68	50.86
	GRASPrune	**62.54**	**59.17**	58.88	60.86	37.20	37.00	**69.80**	55.06
	KAT-Prune	62.32	58.03	**62.43**	**64.35**	**39.08**	36.40	68.99	**55.94**
Qwen2.5-14B	Dense	85.26	82.92	75.30	79.00	58.45	45.60	82.43	72.71
	Wanda-SP	65.84	52.42	55.72	65.66	39.59	36.00	69.59	54.97
	FLAP	65.54	55.50	64.48	62.96	38.99	40.00	67.36	56.40
	Týr	**78.29**	63.75	64.72	**72.94**	**47.10**	40.80	72.42	**62.86**
	GRASPrune	63.73	**68.17**	61.88	66.46	42.83	**41.40**	**73.45**	59.70
	KAT-Prune	62.26	62.23	**64.80**	70.79	45.90	39.80	70.40	59.46

Table 9: Zero-shot downstream accuracy (%) at 50% target pruning.

Model	Method	BoolQ ↑	HellaSwag ↑	WinoGrande ↑	ARC-E ↑	ARC-C ↑	OBQA ↑	PIQA ↑	Avg_7 ↑
Llama-2-7B	Dense	79.45	76.26	69.38	75.59	44.71	44.20	78.62	66.89
	Wanda-SP	51.59	27.95	49.01	29.71	22.44	25.40	52.50	36.94
	FLAP	39.51	41.50	52.88	37.42	26.19	32.40	63.00	41.84
	Týr	38.04	45.56	**53.12**	**43.27**	**26.96**	31.20	61.97	42.87
	GRASPrune	60.12	42.35	51.14	38.13	25.77	31.80	62.19	44.50
	KAT-Prune	**61.28**	**46.04**	52.88	41.67	26.37	**33.60**	**63.76**	**46.51**
Llama-3-8B	Dense	81.83	79.22	73.09	80.72	54.18	45.00	80.74	70.68
	Wanda-SP	45.81	28.23	51.14	31.44	20.65	24.60	53.59	36.49
	FLAP	**57.19**	37.69	53.04	36.41	23.21	29.00	58.98	42.22
	Týr	38.32	40.95	54.62	**40.36**	24.49	**31.40**	60.23	41.48
	GRASPrune	49.57	40.08	53.51	39.60	24.91	30.00	**62.46**	**42.88**
	KAT-Prune	41.04	**44.94**	**55.56**	31.19	**25.26**	29.20	59.41	40.94
Llama-3.1-8B	Dense	83.00	79.34	74.59	82.11	54.86	45.20	80.90	71.43
	Wanda-SP	43.09	28.00	50.91	31.14	20.82	24.60	52.39	35.85
	FLAP	59.24	39.07	54.14	38.34	23.89	29.20	59.25	43.30
	Týr	43.12	42.16	54.22	**39.14**	**26.71**	31.20	62.24	42.68
	GRASPrune	**61.44**	41.60	52.64	39.10	25.85	**32.60**	**62.57**	**45.11**
	KAT-Prune	51.74	**45.15**	**57.22**	36.07	25.51	30.60	61.92	44.03
Llama-3.2-3B	Dense	73.79	74.08	69.14	74.62	46.25	41.00	78.07	65.28
	Wanda-SP	44.62	29.53	50.20	30.68	21.42	26.20	53.70	36.62
	FLAP	39.20	33.48	51.22	34.81	22.10	27.80	56.69	37.90
	Týr	38.32	35.54	**52.41**	38.38	23.89	28.60	59.30	39.49
	GRASPrune	50.64	34.23	51.14	39.18	23.29	**30.80**	58.54	41.12
	KAT-Prune	**53.67**	**38.16**	51.14	**39.94**	**24.06**	27.60	**59.63**	**42.03**
Qwen3-8B	Dense	82.87	78.56	72.53	81.65	56.83	42.20	79.71	70.62
	Wanda-SP	**60.86**	31.87	47.59	27.65	23.38	26.80	53.48	38.80
	FLAP	37.83	35.17	53.12	39.44	23.21	27.40	57.51	39.10
	Týr	40.76	37.23	52.57	38.76	26.96	29.00	58.92	40.60
	GRASPrune	42.42	**43.35**	51.85	**49.28**	**30.63**	**30.40**	**62.62**	**44.37**
	KAT-Prune	56.30	37.76	**53.35**	40.57	23.98	27.40	59.47	42.69
Qwen2.5-14B	Dense	85.26	82.92	75.30	79.00	58.45	45.60	82.43	72.71
	Wanda-SP	57.13	32.94	50.67	39.27	25.17	29.20	57.02	41.63
	FLAP	38.69	36.90	54.06	42.42	26.02	28.60	59.25	40.85
	Týr	56.82	37.25	51.85	**45.71**	26.19	33.20	59.09	44.30
	GRASPrune	**62.05**	**42.45**	**55.41**	44.07	**28.33**	31.20	**62.24**	**46.53**
	KAT-Prune	61.71	40.65	54.06	43.01	25.43	**34.80**	62.02	45.96